

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra měřicí a řídicí techniky

**Návrh pracoviště pro automatickou kontrolu stanovených
znaků jakosti obráběných dílců pomocí kamerového
systému**

**Proposal of the workplace for automatic control of quality
parametr sof worpieces by camera system**

Prohlášení

*Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně.
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.*

Prohlašuji, že jsem byl seznámen s tím, že na moji bakalářskou/diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. - autorský zákon, zejména §35 - užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a §60 - školní dílo.

Beru na vědomí, že Vysoká škola báňská - Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně ke své vnitřní potřebě bakalářskou/diplomovou práci užít (§35 ods. 3).

Souhlasím s tím, že jeden výtisk bakalářské/diplomové práce bude uložen v Ústřední knihovně VŠB-TUO k prezenčnímu nahlédnutí a údaje o bakalářské /diplomové práci budou zveřejněny v informačním systému VŠB-TUO.

Beru na vědomí, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě:

.....
Václav Kreuziger

Poděkování

Chtěl bych touto cestou velmi poděkovat vedoucímu mé diplomové práce prof. Ing. Vilému Srovnalovi za cenné rady, konzultace a připomínky spojené s vypracováním mého úkolu. Dále také firmě Schmachť CZ s.r.o a odtud především panu Milanu Matulovi za zapůjčení inteligentní kamery a v neposlední řadě panu Ing. Ladislavu Hrubému z firmy TEAZ STROJE s.r.o za zapůjčení osvětlení. Bez této podpory bych nemohl dospět k níže popsaným výsledkům.

Abstrakt

Tato diplomová pojednává o použití kamerových systému, zejména pak o tzv. inteligentních kamerách firmy Cognex. Je zde popsána problematika výběru objektivu a osvětlení. Práce popisuje aplikaci inteligentní kamery při výstupní kontrole obráběných dílců. Součástí je také návrh kontrolního stanoviště a řídicího systému.

Klíčová slova

inteligentní kamera, optická kontrola, výstupní kontrola, In-Sight Explorer, Cognex

Abstract

The thesis treats of application cameras systems especially smart cameras Cognex. It is describe choice problem lens and lighting. The thesis describes application smart cameras to verification work piece. It takes in design verification tool and control system.

Key Words

smart camera, optical verification, output verification, In-Sight Explorer, Cognex

Seznam použitých symbolů a zkratek

A/D – analogově digitální převodník

CMOS – typ čipu kamery

CCD – typ čipu kamery

CPU – centrální procesorová jednotka

DO – digitální výstup (digital input)

DI – digitální vstup (digital output)

FOV - Field Of View (zorné pole)

OCR – optické rozpoznání znaků (Optical character recognition)

PC – osobní počítač

PCI, PCIe – typ sběrnice užívané ve výpočetní technice

RAM – operační paměť (Random Access Memory)

LED – světlo emitující dioda

UV – ultrafialové záření

MOSFET – elektrickým polem řízený tranzistor (Metal Oxide Semiconductor Field Effect Transistor)

PLC – programovatelný logický automat

Obsah

OBSAH.....	6
1 ÚVOD	1
2 PŘEDMĚT KONTROLY	2
3 VOLBA KAMEROVÉ SYSTÉMU.....	3
3.1.1 <i>Inteligentní kamery.....</i>	<i>4</i>
4 VOLBA OSVĚTLENÍ.....	8
5 VOLBA OBJEKTIVU	11
6 NÁVRH SESTAVY KONTROLNÍHO STANOVISŤE.....	14
6.1 PŘÍPRAVEK PRO POLOHOVÁNÍ.....	15
6.1.1 <i>Řídící deska krokového motoru</i>	<i>16</i>
6.2 ŘÍDICÍ SYSTÉM POLOHOVACÍHO MECHANIZMU	18
6.2.1 <i>Registrace vadných součástek.....</i>	<i>20</i>
6.2.2 <i>Rozhraní člověk - stroj (HMI)</i>	<i>21</i>
6.3 PROGRAM IN-SIGHT EXPLORER.....	21
6.4 DETEKCE VAD	22
6.4.1 <i>Detekce hran</i>	<i>22</i>
6.4.2 <i>Morfologické operace.....</i>	<i>23</i>
6.4.3 <i>Segmentace prahováním.....</i>	<i>24</i>
6.4.4 <i>Postup nalezení vad na kontrolované součástce</i>	<i>25</i>
6.4.5 <i>Použité funkce</i>	<i>28</i>
6.5 MĚŘENÍ ROZMĚRŮ	31
6.5.1 <i>Metrické vlastnosti obrazu</i>	<i>31</i>
6.5.2 <i>Užití smart kamery pro měření rozměrů</i>	<i>31</i>
6.5.3 <i>Kamery Cognex In-Sight</i>	<i>32</i>
6.5.4 <i>In-Sight 5100C</i>	<i>33</i>
6.5.5 <i>In-Sight 1403.....</i>	<i>33</i>
6.5.6 <i>Program pro měření rozměrů.....</i>	<i>33</i>
6.6 RYCHLOST KONTROLY	34
6.7 SPRÁVA UŽIVATELŮ	34
7 ZÁVĚR.....	36
8 POUŽITÁ LITERATURA	37
9 SEZNAM PŘÍLOH	38

1 Úvod

V dnešní době dosahují výpočetní systémy poměrně vysokého výkonu. Což umožňuje řešit úlohy náročné na objem zpracovávaných dat jako je analýza a zpracování obrazu. Současně vysoká integrace elektronických obvodů (např. CPU, řadiče, paměti,...) těchto systémů dovoluje vytvořit kamerový systém, který obsahuje nejen vlastní kameru, ale také hardware pro zpracování a analýzu obrazu. Tudíž je snaha ze strany výrobců, jejichž produkty mají dosahovat vysoké jakosti výroby, nahrazovat lidskou sílu při kontrole, manipulaci nebo třídění těmito kamerovým systémy. A to zejména tam, kde je dosahováno vysoké sériovosti výroby a výrobek je určen např. pro automobilový průmysl. Důvody jsou často monotonií a neustále se opakující činnost operátora, který provádí např. kontrolu obrobku. Při takové kontrole nelze zcela přesně zajistit opakovatelnost a tudíž ani stálou přesnost a kvalitu. Ta je dána především morálkou operátora a jeho fyzickým a psychickým stavem.

Tato práce pojednává o použití kamerových systémů (především inteligentních kamer) při kontrole kvality výroby, volbě osvětlení, objektivu a kamery. Ke kontrole nebyl vybrán zcela typický typ objektu určeného pro řešení pomocí těchto prostředků. Výsledkem by tedy mělo být také ověření jejich možností.

Kamerové systémy se v současnosti využívají např. kontrole etiket, sestavení výrobku, čtení čárových, 2D kódů a ve spojení s manipulátorem ke tvorbě inteligentních polohovacích a výrobních prostředků.

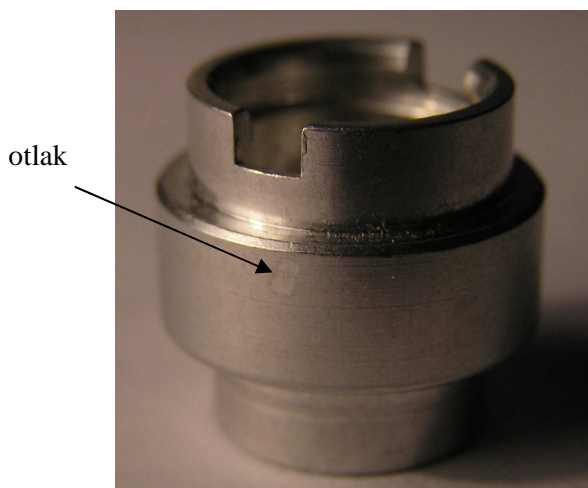
V druhé kapitole je popsán předmět kontroly, úskalí detekce a možné komplikace. Následující kapitola pojednává o volbě kamerového systému, rozdíl mezi inteligentní kamerou a normální kamerou. Je zde uvedena i tabulka s některými kamerami různých výrobců. Volba osvětlení a objektivu je popsána v kapitolách 4 a 5. V kapitole 6 popisují konkrétní řešení zadaného problému. Což je detekce otlaku a kontrola rozměrů. V poslední kapitole jsem se pokusil shrnout dosažené výsledky.

2 Předmět kontroly

Předmětem kontroly je součástka vyráběná pro automobilový průmysl. Což je také důvodem snahy o co nejvyšší kvalitu výstupního výrobku. V současné době je kontrola prováděna operátorkami manuálně.

Jedná se o hliníkový obrobek, na kterém se v průběhu výroby mohou vytvářet nežádoucí otlaky nebo drážky. Otlaky jsou nejčastěji způsobeny přichyceným materiálem na upínacích čelistech obráběcího stroje, které do obrobku vytlačí prohlubeň. Drážky způsobuje špatný obráběcí nůž. Jednou z dalších vad výrobku je zachycení brousícího tělíska v jeho těle. Tyto vady je nutno na výrobku detekovat a oddělit od bezvadných. Kamerový systém musí také změřit důležité rozměry v zadané přesnosti 0.1 mm .

Kontrovaná součástka je zobrazena na obr. 1. Je zde vidět jeden z možných otlaků. Vzhledem k tomu, že tvar a umístění otlaků na kontrolovaných plochách je náhodné, není možné navrhnout algoritmus kontroly založený na hledání předdefinovaných tvarů vad, jak by se mohlo nabízet. Další komplikací je materiál, ze kterého je součástka vyrobena. Jedná se o slitinu hliníku a ta se vyznačuje vysokou odrazivostí. Je tedy nutné zvolit vhodný zdroj, úhel a intenzitu osvětlení. Vzhledem k velikosti součástky bude problém jeho polohování před kamerou.



Obr. 1 Kontrovaná součást

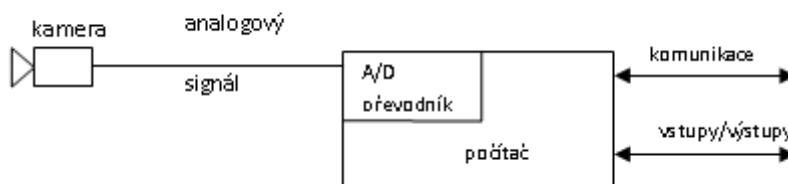
3 Volba kamerové systému

Kamery a kamerové systémy lze rozdělit podle způsobu zpracování a analýzy obrazu na dva základní druhy. Jedním způsobem řešení je tzv. inteligentní kamera, což je vestavěný systém určený pro zpracování určité oblasti aplikací zpracování a analýzy obrazu (např. kontrola přítomnosti součástí výrobku, správné orientace, čtení čárového kódu). Podle rozsáhlosti a náročnosti aplikace jsou k dispozici jednoduché inteligentní kamery, jejichž konfigurace je velice snadná, jsou určeny pro ty nejzákladnější aplikace. Výrobci těchto systémů je přirovnávají k inteligentním čidlům. Konfigurace těchto kamer se provádí pomocí rozhraní přímo na těle kamery (klávesnice, display) nebo softwarového nástroje v PC.[3][4][5]

Pro náročnější úlohy jsou určeny inteligentní kamery, které disponují již větším množstvím sofistikovanějších nástrojů pro zpracování obrazu, nechybí zde např. různé způsoby detekce hran, 2D kódu, vzoru, matematické operace, nástroje pro komunikaci, atd. Ke tvorbě programu je používáno PC a speciální vývojový software. Ke komunikaci mezi PC nebo PLC je v poslední době používán výhradně Ethernet.[3][4][5]

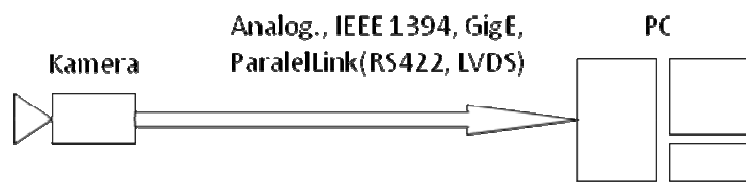
Jiným způsobem řešení kamerového systému je použití spojení kamery s PC, průmyslovým PC, emdeded PC, atd. Toto řešení je používáno pro náročnější aplikace, kde by inteligentní kameru nebylo možné použít. Jednak je to způsob finančně náročnější, protože je nutné zakoupit jednak hardware jako je kamera, PC s potřebným rozhraním tak software pro zpracování obrazu. Na druhou stranu dostaneme nástroj, kterým je možno řešit širokou škálu aplikací na zpracování obrazu. Ke komunikaci mezi kamerou a PC se využívá standardních rozhraní jako FireWire, USB2, Gigabit Ethernet nebo digitální paralelní port standardu RS422 nebo LVDS. Tato rozhraní jsou většinou dodávány jako zásuvné karty do PCI nebo PCIe slotu PC.[6]

Blokové schéma strojového vidění je zobrazeno na obr. 2. V případě inteligentní kamery jsou všechny bloky součástí hardwaru kamery. Jinak je kamera tvořena čipem s A/D převodníkem a některým výše popsaným rozhraním se kterým komunikuje s PC.[4][5]

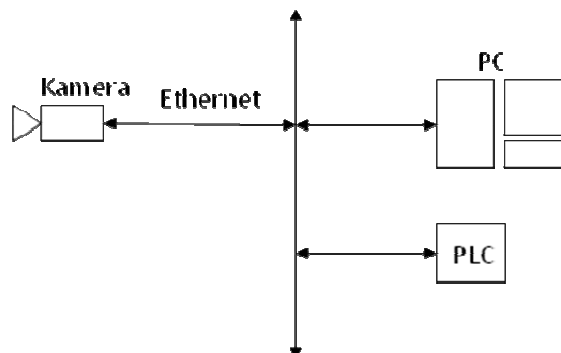


Obr. 2 Schéma strojového vidění

Vzhledem ke snaze minimalizace finančních nákladů je primárně snaha využít k řešení inteligentní kameru. V současné době však někteří výrobci nabízejí kamerové systémy o vysokém rozlišení, jejichž cena je výhodnější než tzv. inteligentních kamer. Vyplatí se pořídit kameru, software a PC při zachování poměrně výhodné ceny. Důvodem je, že inteligentní kamery s vysokým rozlišením (řádově 1200x1600 bodů) jsou zatím drahé. Uspořádání komunikace kamerového systému, u něhož probíhá analýza v PC je zobrazeno na obr. 3, komunikace inteligentního kamerového systému je na obr. 4.



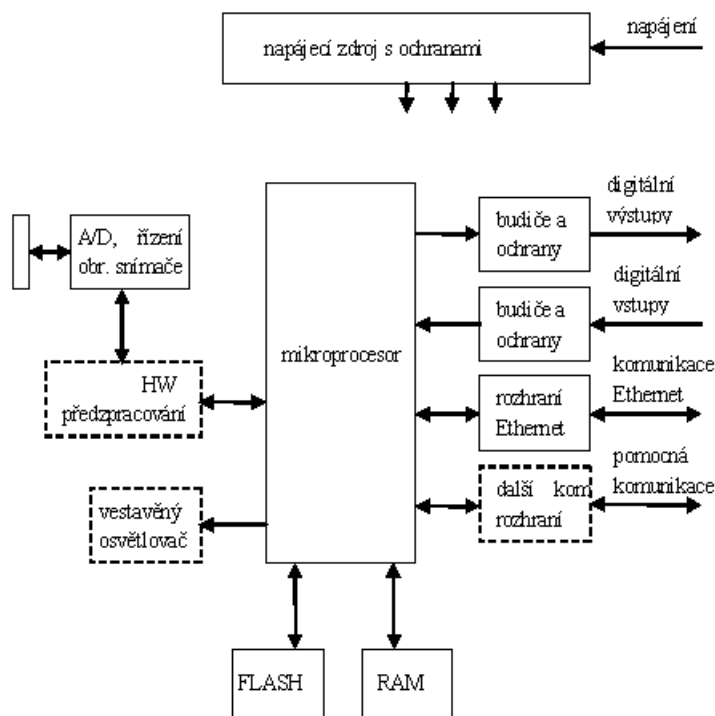
Obr. 3 Komerový systém (analýza obrazu probíhá na PC)



Obr. 4 Komerový systém (analýza obrazu probíhá na HW kamery)

3.1.1 Inteligentní kamery

Současný vývoj na poli elektroniky dovolil, že dochází k integraci vlastní kamery s procesorem a pamětí do jednoho systému. Takový systém je nazýván inteligentní kamerou v anglosaské literatuře smart camer. Jedno nebo více rozměrný signál pořízený kamerou nebo systémem kamer je dále zpracováván na požadovanou informaci o stavu sledovaného objektu např. o tom, zda se nachází ve sledované oblasti, o jeho orientaci, směru a rychlosti pohybu, barvě, tvaru nebo rozměru. Objem dat získaných kamerou je obvykle podstatně vyšší, než je požadováno uživatelem, a proto jsou data redukována pouze na ta, která jsou vhodná k dalšímu zpracování. V podstatě se jedná o vestavěný systém, jehož součástí je kamera, řídicí jednotka, vyhodnocovací jednotka a komunikační rozhraní. Toto rozhraní slouží ke komunikaci s řídicím systémem např. PLC, PC, ... Uživatelem vytvořený program zpracování obrazu je uložen v paměti inteligentní kamery a je zpracováván vlastním hardwarem kamery. Jelikož zpracování obrazu vyžaduje zpracování velkého množství dat, jsou využívány výkonné procesory většinou DSP. Velikost paměti RAM je především dána velikostí uživatelského programu, který se po inicializaci kopíruje z programové paměti typu FLASH do RAM a zde je spuštěn. Paměť FLASH bývá většinou vestavěná. Mimo zpracovávaného programu je v paměti RAM uložen i celý obraz snímané scény. K urychlení provedení požadovaných operací (např. detekce hran, filtrace,...) se někdy používají konečné automaty programované v hradlových polích. Výstupem takové kamery nemusí být digitální ani analogový obrazový signál ale analogová nebo digitální hodnota určující sledovaný stav nebo transformaci snímaného obrazu. Jedná se o snahu přiblížit inteligentní kamery běžným senzorům. Digitální výstupy jsou většinou typu NPN nebo PNP s otevřeným kolektorem s pracovním napětím do 30V s ochranami proti přepólování a poškození napětíovou špičkou při spínání indukční zátěže. Digitální vstupy jsou využívány pro synchronizaci sejmoutí snímku se stavem procesu. Kontrolovaný objekt je většinou nutné sejmout v určité poloze a kamera se spouští například přibližovacím čidlem nebo signálem z řídicího systému. [4][5]



Obr. 5 Blokové schéma smart kamery

V následujících odstavcích budou podrobněji popsány jednotlivé součásti inteligentní kamery.

3.1.1.1 Část snímání a digitalizace

Do nedávné doby byl nejčastěji používaným obrazovým senzorem v kamerách snímač CCD. Poskytuje v principu analogový signál, který musí být digitalizován převodníkem A/D. Celý proces snímání bývá řízen u jednodušších kamer přímo výpočetní jednotkou, u výkonnějších kamer jsou snímání a digitalizace řízeny zvláštním mikropočítačem nebo hardwarovou řídicí jednotkou, která je z výpočetní části pouze synchronizována. V poslední době došlo ke značnému pokroku v technologii obrazových senzorů CMOS, které jsou již schopny poskytnout obraz kvality srovnatelné se snímačem CCD. Technologie CMOS navíc umožňuje integrovat na snímací čip i převodník A/D a většinu řídicích obvodů. Obsluha snímače CMOS je podstatně jednodušší, podobá se vybírání dat z dynamické paměti RAM. Toto zjednodušení je při konstrukci smart kamery vítané, neboť dochází ke zjednodušování a zlevňování kamery s obrazovým snímačem CMOS. [4][5]

3.1.1.2 Výpočetní část

Výpočetní část smart kamery je vlastně mikropočítač. Protože zpracování obrazu vyžaduje zpracování velkého množství dat, bývají použity výkonné mikroprocesory a vzhledem k charakteru výpočtu často i digitální signálové procesory (DSP). [4][5]

3.1.1.3 Komunikační rozhraní

Komunikační rozhraní plní u smart kamer několik funkcí. Především se jedná o připojení k rozhraní MMI (Man-Machine-Interface)-zařízení, které umožňuje kameru nastavovat a případně programovat. V současné době slouží jako rozhraní jako rozhraní MMI nejčastěji standardní počítač typu PC. Pro pohodlný vývoj aplikace je třeba, aby zařízení MMI zachycovalo obraz snímáný

kamerou v reálném čase. Přenosová rychlost pak musí být poměrně vysoká, a proto se jako základní komunikační rozhraní používá nejčastěji Ethernet.

Komunikační rozhraní může také přenášet data do nadřazeného řídicího systému nebo pomocí modulu umožňuje rozšířit počet vstupů a výstupů. V neposlední řadě může také zprostředkovat komunikace mezi několika kamerami v případě složitější úlohy, která vyžaduje spolupráci několika kamer. Většinou slouží i k zajišťování servisních služeb, výměně firmwaru kamery a k podobným účelům. Někdy bývá smart kamera kromě obvyklého rozhraní Ethernet vybavena ještě pomocným sériovým rozhraním RS-232/422/485. Důvodem je snadnější připojení k většině existujících PLC. Vyšší typy průmyslových sběrnic a protokolů (Profibus a Modbus) se u smart kamer zatím běžně neuplatňují, v případě potřeby se používá externí komunikační převodník. [4][5]

3.1.1.4 Provedení smart kamer

Mechanické provedení smart kamer není dosud standardizováno, jako je tomu například u běžných snímačů. Smart kamera se nejčastěji podobá standardní průmyslové kameře používané v klasických systémech strojového vidění. Liší se množstvím připojovacích míst, má navíc vstupy a výstupy i komunikační rozhraní. Mnoho výrobců doplňuje smart kameru i vestavěným kruhovým osvětlovačem s LED. Nezávislost kamery je tím sice dovršena, avšak jednoduchý vestavěný osvětlovač je vhodný pouze pro nejjednodušší aplikace a ve většině případů je nedostačující. V posledních letech umožnila pokračující miniaturizace kvalitativní zlom v konstrukci smart kamer, které se začaly montovat do pouzder standardních přibližovacích senzorů, což je projevem dalšího přibližování strojového vidění k senzorové technice. [4][5]

3.1.1.5 Software a programování

Smart kamera není zařízení, do kterého lze instalovat libovolný software. Toto omezení je dáno úzkou vazbou na technické prostředky – mikroprocesor, omezený rozsah paměti, způsob připojení snímacího čipu i vstupu a výstupu. Software instalovaný ve smart kameře tedy ve velké míře určuje i oblast aplikací, pro které je kamera určena. Nejlépe je to patrné na vysoce specializovaných smart kamerách určených pouze pro jednu třídu úloh. Typickou ukázkou jsou kamery, které umí pouze porovnat snímáný objekt s naučeným vzorem. K porovnání nejčastěji dochází na základě podobnosti zjišťovaných obrysů nebo na základě podobnosti jasového histogramu v zadané oblasti. Programování takové kamery je poměrně jednoduché. Po instalaci kamery sejme správný objekt a nastavíme hranice (nejčastěji pomocí jednoho nebo několika koeficientů), ve kterých lze ostatní objekty ještě považovat za správné. Ačkoliv se jedná o velmi jednoduchý princip analýzy obrazu, je překvapivé, že pro řešení velkého počtu je plně postačující. [4][5]

Software pro univerzální smart kamery využívá stejné principy jako software velkých systémů strojového vidění. Principem je rozdělení celkové úlohy do podúloh, které lze řešit standardizovanými softwarovými nástroji. Mezi pod úlohu patří například hledání obrysů, hledání uzavřených tvarů a zjištění jejich polohy a velikosti (blob recognition), hledání odlišností (flaw detection), čtení písma (OCR – Optical character recognition) a měření vzdálenosti. [4][5]

Někteří výrobci dodávají uvedené nástroje ve formě knihoven funkcí pro obecný programovací jazyk (nejčastěji C). Uživatel si sám a s využitím těchto funkcí napíše program pro danou úlohu, přeloží ji pro specifický procesor používaný smart kamerou, a poté ji zavede do programové paměti kamery. Dodavatelé doplňují knihovny, simulátory a dalšími prostředky pro usnadnění vývoje programu, výsledek však bývá závislý především na schopnostech programátora.

Pokročilejší systémy využívají vývojového systému pracujícího na připojeném PC, který spolupracuje s firmwarem kamery a umožňuje vizuální parametrizaci nástrojů i sestavení finálního

programu ve formě vývojového diagramu nebo tabulky popisující proces zpracování obrazu pomocí posloupností nástrojů. Vývoj aplikace je velmi jednoduchý. Aplikace však nebývá optimalizována z hlediska rychlosti běhu, což uživatele často nutí využít výkonnější a tím i dražší kameru. [4][5]

3.1.1.6 Typy inteligentních kamer na českém trhu podle výrobců

Na českém trhu lze nalézt několik distributorů inteligentních kamer značek např. Cognex, Siemens, Sick, Balluff, Datasensor, National Instruments a další. Jejich sortiment lze většinou rozdělit na kamery nižší a vyšší třídy, podle rozsahu softwarových nástrojů a z toho možností jejich použití při řešení konkrétní aplikace. Vzhledem k předpokládané náročnosti řešené úlohy se soustředím pouze na kamery vyšší třídy.[2][7][8]

Důležitými vlastnostmi, které by měla použitá kamera splňovat, jsou potřebné rozlišení, komunikační rozhraní a knihovna potřebných softwarových nástrojů pro zpracování a analýzu obrazu. Velikost rozlišení ovlivňuje výslednou přesnost měření rozměrů součástky, a protože je potřebná integrace kamerového systému s řídicím systémem kontrolního stanoviště, je nutné, aby disponovala komunikačním rozhraním. Barevné rozlišení není důležité již z tohoto důvodu, že pro řešení úlohy není potřeba rozlišovat barvy. Postačovala by tedy černobílá kamera. Rychlost kontroly není příliš kritická, takže počet snímků, které kamera sejme za sekundu, nerozhoduje.

Jelikož firma, pro kterou je tato diplomová práce zpracovávána má partnerské vztahy s firmou Schmachtl s. r. o, která je distributorem inteligentních kamer Cognex, byla vybrána inteligentní kamera In-Sight 5100C.

Následující tabulka 1 uvádí některé typy kamer a jejich parametry pro srovnání.

Tabulka1: Parametry některých inteligentních kamer

Výrobce a typ	Cognex In-Sight 5100C	Cognex In-Sight 5403	Sick IVC-2D	Siemens VS723A
Senzor	1/3 palce CCD	1/1.8 palce CCD	výrobce neudává	1/3 palce CCD
Rozlišení	640 x 480 bodů	1600 x 1200 bodů	640 x 480 bodů	640 x 480 bodů
Počet barev	24b	256 stupňů šedi,	barevná	černobílá
Počet snímků za sekundu	42 snímků plného rozlišení	13 snímků plného rozlišení	výrobce neudává	60 snímků plného rozlišení
Vstupy/Výstupy	externí spouštění, externí vstupy a výstupy řešené pomocí rozšiřujících modulů, dva vestavěné rychlé výstupy, výstup pro řízení externího osvětlení	Tato kamera disponuje totožnými parametry jako In-Sight 5100C	externí spouštění, 3 programově řízené výstupy a 4 programově řízené vstupy, výstup pro řízení externího osvětlení	vstupů 32 a 64 výstupů přes rozhraní ethernet
Komunikační rozhraní	Ethernet, RS232C		Ethernet, RS485	průmyslový ethernet
Napájení	24VDC 350mA max.		DC 19.2 ... 28.8 V 400mA	24V DC 300mA
Krytí	IP67		IP67	IP 51

4 Volba osvětlení

Volba vhodného osvětlení tvoří asi nejdůležitější část strojového vidění. Hlavním cílem je dosáhnout maximálního kontrastu mezi cílovým objektem a pozadím. Zdroj světla musí poskytnout jednotné světelné podmínky, které nebudou ovlivňovány okolním prostředím. Je nutné zajistit, aby okolní světlo nedopadalo na snímanou scénu. To provedeme volbou světla, které je výrazně jasnější než okolí a tento zdroj umístíme v blízkosti součástky, případně použijeme filtr potlačující okolní světlo. Lze samozřejmě prostor, ve které se provádí optická kontrola zakrýt, to však sebou nese horší přístupnost. Jednoduše založit procesní analýzu na vysokém kontrastu mezi dobrou a špatnou součástí.[3]

Důležitými faktory při výběru osvětlení jsou optické vlastnosti snímané součásti (barva, tvar, povrch, odrazivost), typ osvětlení (boční, zadní), geometrie osvětlení (geometrické vlastnosti světelného zdroje), zdroje světla (LED, halogenové, fluorescentní osvětlovače). [3]

Většina kamerových systémů získává obraz pouze ve stupních šedi, což v podstatě odpovídá intenzitě odraženého světla. Potom je poměrně důležité znát spektrum světla vyzařovaného jeho zdrojem. [3]

Běžně používanými zdroji světla jsou LED, fluorescentní a halogenové osvětlovače. LED osvětlovací jednotky se běžně vyrábí v barvách RGB, IRF a bílé světlo. Používají se většinou k osvětlování malých objektů. Dají se použít k vyzdvižení jedné barvy nad ostatní. Aplikujeme-li infračervený osvětlovač spolu s IRF propustným filtrem lze tak potlačit vliv okolního světla. Za jejich výhody považujeme vysokou životnost, odolnost, efektivitu, snesou krátké přetěžování, vhodné k osvětlení malých objektů, nízké tepelné vyzařování. Nevýhodou je jejich poměrně malá intenzita záření a nejsou vhodné pro osvětlování velkých objektů.[3]

Fluorescentní osvětlovače jsou založeny na principu zářivek. Kdy výboj v plynu produkuje UV záření a pomocí fluorescentu (fosfor) je toto záření přeměněno na viditelné světlo. K dostání jsou zdroje bílého světla a UV záření. Používají se převážně k tvorbě velkých rovnoměrně osvětlených ploch. Výhodou je, že jsou vyráběny v mnoha velikostech a tvarech, mohou pokrýt velké oblasti, snadná montáž, cena a přirozený rozptyl. Nevýhodou je mírné oteplení tohoto zdroje světla za provozu, kratší životnost, postupné slábnutí intenzity světla a relativní křehkost. [3]

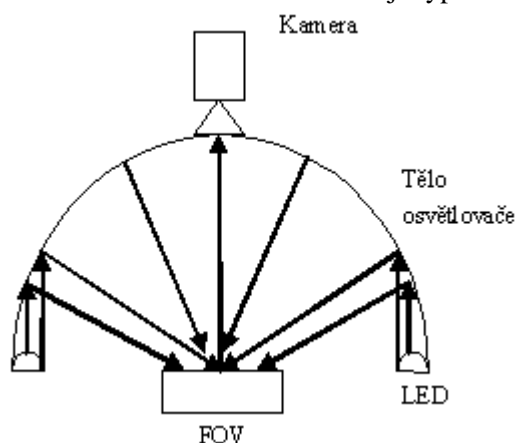
Halogenové zdroje světla jsou tvořeny wolframovým vláknem uzavřeným v baňce naplněné malým množstvím halogenu. Barva vyzařovaného světla je bílá a používají se pro osvětlení jak velkých tak malých oblastí. Jejich použití je upřednostňováno tam, kde je potřeba osvětlit malou plochu velmi intenzivním světlem. Ve spojení s vláknovou optikou je možné světlo směřovat do potřebných míst. Halogenové osvětlovače představují velmi efektivní zdroj bílého světla s velmi vysokou intenzitou. Nevýhodou je však vysoká cena a vyzařované teplo. [3]

Vzhledem k velikosti FOV se jeví jako nejlepší řešení použití LED. Intenzita tohoto osvětlení by měla být dostatečná pro potřeby snímání. Vzhledem k faktu, že analyzovaná součástka je vyrobená z hliníku, není volba barvy příliš kritická. Vybral jsem tedy červené světlo, které se používá nejčastěji.

Důležitou vlastností výběru je také geometrie osvětlení. Což představuje vztah mezi tím, jak se světelné paprsky odráží od povrchu snímaného objektu do objektivu kamery. Vhodnou volbou pozice světelného zdroje lze odstranit odlesky a minimalizovat nedůležité části. Vzhledem k tomu, že se jedná o součástku vyrobenou z kovu, je možné použít difuzní zdroj světla nazývaný *DOME* nebo světelný zdroj označovaný *Low-Angle Light* (světlo které je vyzařováno pod nízkým úhlem). Na obr. 6 je zobrazen příklad realizace světelného zdroje typu *DOME* a na obr. 7 je znázorněn jeho princip. [3]



Obr. 6 Jedna z realizací světelného zdroje typu DOME

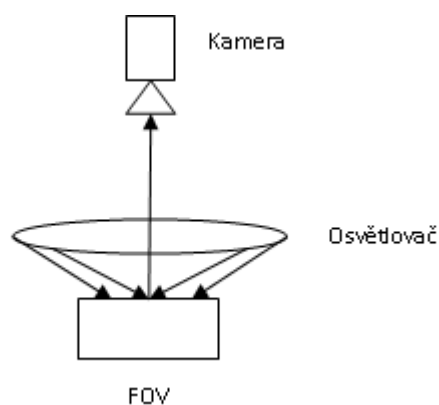


Obr. 7 Princip funkce světla DOME

Na obr. 8 můžeme vidět možnou realizaci druhého typu osvětlení *Low-Angle Light* a na obr. 9 je znázorněn princip funkce.

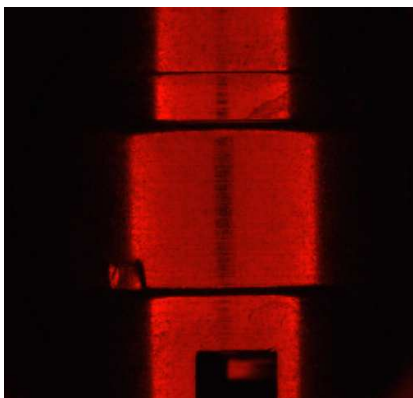


Obr. 8 Jedna z realizací světelného zdroje typu Low-Angle Light



Obr. 9 Princip funkce světla *Low-Angle Light*

K dispozici bylo světlo typů *DOME* a tudíž jsem zvolil toto. Světlo typu *Low-Angle Light* jsem neměl možnost vyzkoušet, proto nemůžu zhodnotit, zda by jeho použití spělo k lepším výsledkům. Následující obrázek představuje kontrolovanou součástku osvětlenou světlem typu *DOME* při pohledu z vrchu a boku. Ta je umístěna na černém podkladu a jsou zde vidět poměrně zřetelně hrany tvaru součástky a otlak (obr. 10).



Obr. 10 Součástka osvětlená světlem typu *DOME*

Při osvětlení jiným typem zdroje světla se na povrchu součástky vyskytovaly poměrně značné odlesky, které by znemožňovaly další analýzu obrazu.

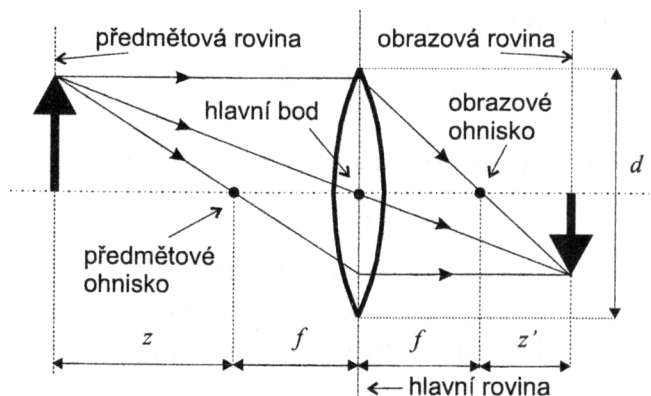
5 Volba objektivu

Vstupní branou obrazu je objektiv, jeho úlohou je promítnout světelné záření na snímač, který převádí ozáření na elektrický signál. Objektiv by měl co nejvěrněji napodobovat ideální projektivní zobrazování. [1][2]

Objektiv je tvořen soustavou několika čoček a clony. Objektivy používané pro průmyslové účely lze rozdělit na tři základní druhy. Normální, kdy je clona před obrazovým ohniskem objektivu, telecentrický u něhož je clona umístěna v obrazovém ohnisku a hypercentrický u něhož je clona umístěna mezi obrazovým ohniskem a obrazovou rovinou. Pro měření rozměrů by byl nejvhodnější telecentrický objektiv u něhož nedochází ke zkreslení perspektivy. Avšak cena těchto objektivů je poměrně vysoká. Kamera, kterou používám, disponuje softwarovým nástrojem, jež slouží k potlačení tohoto zkreslení, tudíž použiji běžný normální typ objektivu. [1][2]

Ideální model dírkové komory více méně platí pro ideální tenkou čočku, která je nejjednodušším objektivem. Projekce paprsku se řídí stejnými zákony jako u dírkové komory a čočka současně sbírá díky konečné apertuře i konečné množství světelné energie (fotonů). [1]

Mějme optickou soustavu jako na obr. 9, kde f je ohnisková vzdálenost, tj. vzdálenost mezi hlavním bodem a (obrazovým, předměťovým) ohniskem, z' je vzdálenost mezi obrazovým ohniskem a obrazovou rovinou, z je vzdálenost mezi předměťovým ohniskem a předmětovou rovinou. Nechť X je velikost předmětu v předmětové rovině (délka šipky) a x velikost obrazu předmětu v obrazové rovině.



Obr. 9 Tenká čočka

Zvětšení optické soustavy m je dáno:

$$m = \frac{x}{X} = \frac{f}{z} = \frac{z'}{f} \quad (1)$$

V následujících odstavcích bude znázorněn postup příkladu řešení. Použitá kamera má čip o rozměru $1/3''$ čemuž odpovídají rozměry $4.74\text{mm} \times 3.55\text{mm}$. Jelikož snímaná součást má tvar válce, jehož průměr je 17mm a výška 16mm , zvolím velikost snímaného pole (*ang. FOV field of view*) kamerou $20\text{mm} \times 20\text{mm}$. Již fakt, že čip má obdélníkový tvar, nebude tyto rozměry možné dodržet. Řekněme tedy, že jedna strana FOV bude mít rozměr 20mm a bude na čipu kamery odpovídat straně o délce 4.74mm . Potom zvětšení objektivu bude:

$$m = \frac{x}{X} = \frac{4.74}{20} = 0,237 \quad (2)$$

Pokud použiji nejběžnější objektiv s ohniskem 25mm , pak bude nutno použít distanční kroužek o výšce:

$$z' = m \cdot f = 0,237 \cdot 25 = 5,75 \text{ mm} \quad (3)$$

Nejmenší vyráběné velikosti distančních kroužků jsou 5mm a 1mm. Použiji tedy kroužek 5mm a podle potřeby 1mm. Tím získám distanci 5mm nebo 6mm.

Vzdálenost z od snímané scény bude:

$$z = \frac{f}{m} = \frac{25}{0,237} = 105,48 \text{ mm} \quad (4)$$

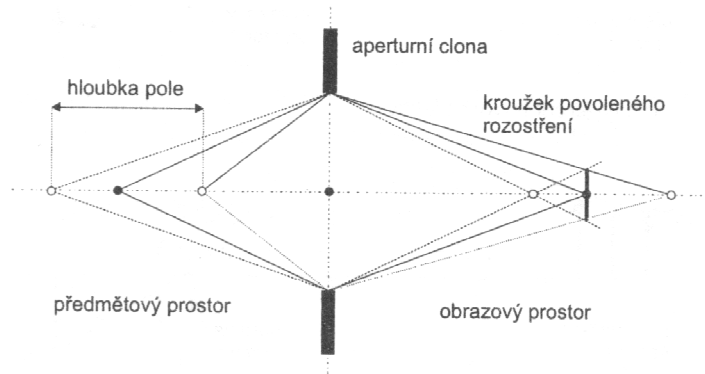
Poměrně důležitým parametrem je hloubka ostrosti (angl. depth of focus) někdy označována jako hloubka pole (angl. depth of field). Vychází s představy, že v aplikacích je přípustné, aby obraz byl poněkud rozostřený. Pro digitální obrazy je povolené rozostření úměrné velikosti pixelu. Čím je rozlišení větší, tím méně může být obraz rozostřen. Hloubkou ostrosti je interval povolených posunů obrazové roviny $\langle -\Delta z, \Delta z \rangle$, pro který je průměr kruhu rozostření menší než předdefinovaná hodnota. Hledaná hodnota Δz se vypočte podle vzorce:

$$\Delta z = n_f \cdot \left(1 + \frac{f}{z'}\right) \cdot \varepsilon = n_f \cdot (1 + m) \cdot \varepsilon \quad (5)$$

Kde n_f představuje f -číslo objektivu:

$$n_f = \frac{f}{d} \quad (6)$$

Tato rovnice ukazuje, jakou roli hraje f -číslo objektivu pro hloubku ostrosti. Z výše uvedeného vzorce je vidět, že čím větší je průměr clony tím menší je f -číslo objektivu a tím pádem menší hloubka ostrosti. Pokud tedy chceme zvětšit hloubku ostrosti, pak musíme volit menší průměr clony, což ale negativně ovlivní množství dopadajících fotonů na čip kamery (osvětlení čipu). Čímž bychom získali ne příliš kvalitní, málo kontrastní obraz. Je proto nutné nalézt vhodný kompromis mezi ohniskem objektivu a číslem clony. Samozřejmě platí, že čím větší je ohnisko objektivu, tím větší je hloubka ostrosti. [1]

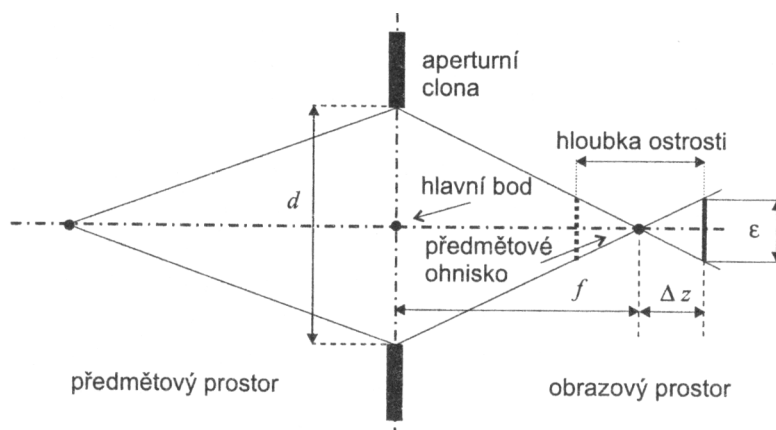


Obr. 11 Hloubka pole na předmětové straně optické soustavy

Proměnná ε je hodnota průměru kruhu, na kterou se zobrazí bod v předmětovém prostoru. Celá situace je zobrazena na obr. 12

Průměr kruhu ε se vypočte podle podobných trojúhelníků:

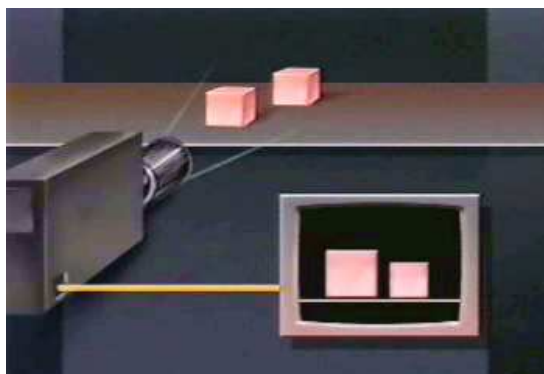
$$\begin{aligned} \frac{d}{f + z'} &= \frac{\varepsilon}{\Delta z} \\ \varepsilon &= \frac{d \cdot \Delta z}{f + z'} = \frac{\Delta z \cdot f}{n_f \cdot (f + z')} \end{aligned} \quad (7)$$



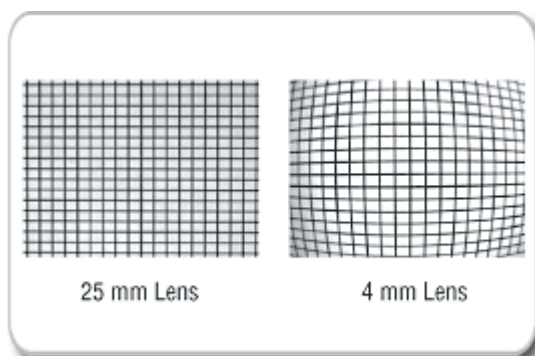
Obr. 12 Hloubka ostrosti v obrazovém prostoru optické soustavy

Hodnota hloubky ostrosti poměrně důležitá při měření rozměrů trojrozměrných předmětů. Je důležité, aby interval $\langle -\Delta z, \Delta z \rangle$ měl velikost shodnou s výškou měřeného objektu. Pokud by tomu tak nebylo došlo by k nežádoucímu rozmazání hran a tím pádem k nepřesnosti měření. Tento interval je většinou určen velikostí clony ručně, vizuální kontrolou. [1]

Při výběru objektivu je také nutné věnovat pozornost zkreslení, které může být zkreslení perspektivy a geometrické. Zkreslení perspektivy lze ovlivnit volbou typu objektivu. Pokud je nutné, aby k tomuto zkreslení nedocházelo, je vhodné použít objektiv telocentrický. Použití tohoto objektivu je výhodné např. při měření rozměrů 3D objektů nebo jsou-li měřené rozměry od objektivu v jiné vzdálenosti. Geometrické zkreslení lze ovlivnit volbou ohniska objektivu. Platí pravidlo, že objektivy s menší ohniskovou vzdáleností mají větší toto zkreslení vyšší. Příklad zkreslení perspektivy je vidět na obr. 13 a geometrické zkreslení na obr. 14.



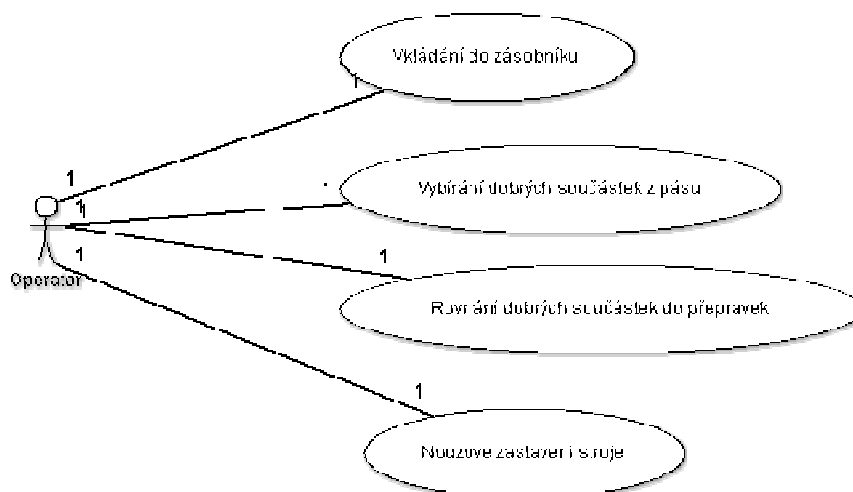
Obr. 13 Příklad zkreslení perspektivy



Obr. 14 Deformace způsobená ohniskovou vzdáleností objektivu

6 Návrh sestavy kontrolního stanoviště

Vzhledem k povaze výroby kontrolované součástky, která je rozdělena na několik částí, není kontrolu zcela zautomatizovat. Součástka je vyráběna na obráběcích centrech a manipulaci mezi jednotlivými stupni výroby zajišťuje člověk. Nejedná se tedy o provoz se zcela automatickou výrobou, kde by nebylo potřeba zásahu člověka. Z tohoto důvodu je kontrolní stanoviště řešeno jako samostatná jednotka ve výrobním procesu, na konci řetězce. Kontrolované součástky jsou k tomuto stanovišti dopravovány operátorem, který má na starosti předchozí operaci. Přístup operátora k samotnému stroji je zobrazen na obr. 15. za pomoci diagramu případů užití.



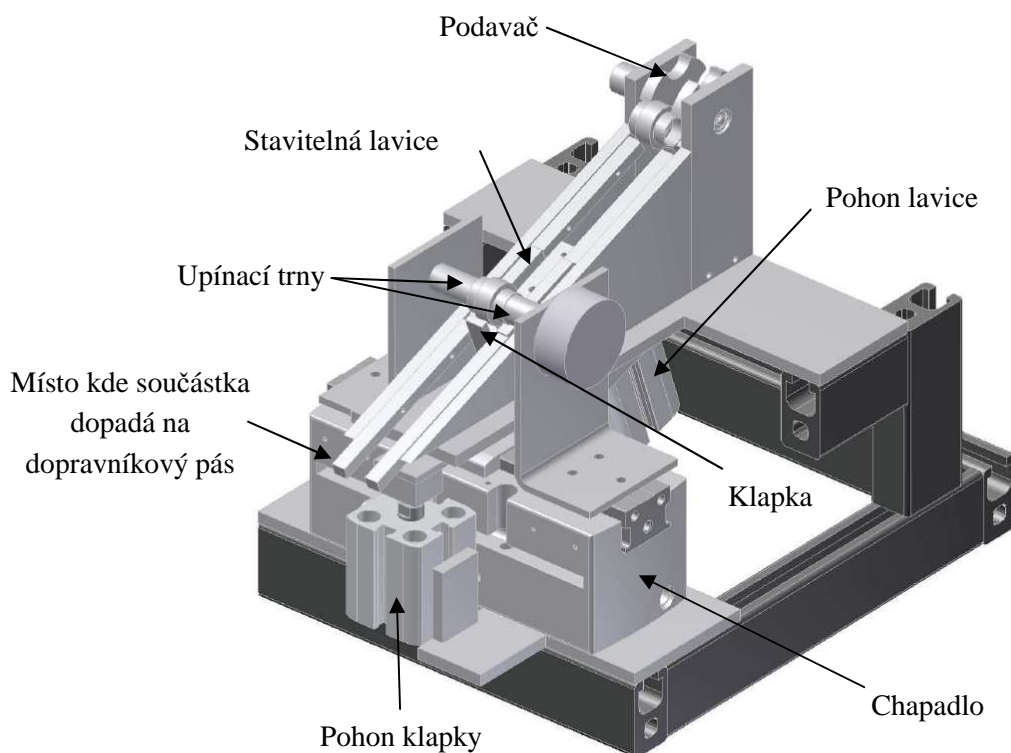
Obr. 15 Přístup operátora k třídícímu zařízení

Jak je vidět na obrázku výše úkolem operátora je doplňovat zásobník se součástkami, přemísťovat bezvadné součástky do přepravek a v případě poruchy zastavit činnost stroje. Pozice operátora při kontrole součástek bude zachována, ale dojde ke zvýšení její přesnosti vlivem automatizace. Operátor bude ušetřen činnosti kontroly náročné na soustředění. Kontrola rozměrů je prováděna statisticky, kdy je několikátá součástka přeměřena a pokud jsou rozměry v dané toleranci, není třeba zavádět žádná opatření.

Jak již bylo psáno výše, bude se jednat o samostatné zařízení, je tedy nutné aby obsahovalo vstupní zásobník, do kterého operátor vloží určitý počet součástek. Ten bude nutně obsahovat pohon, pomocí něhož budou součástky dávkovány před kameru. Samotnou dopravu součástky do prostoru kamery lze řešit několika způsoby. Důležité je, aby každá součástka byla v neměnné pevně dané pozici před objektivem kamery. Nelze tedy použít běžný pásový dopravník. Vhodným způsobem je požití vodících lyžin. Komplikací je také tvar součástky, který je válcový s funkčními plochami.

Po vodících lyžinách je součástka dopravena před kameru a zde upnuta do trnů chapadla. Součástka je zastavena pomocí klapky před těmito trny. Trny jsou upevněny na kleštinách pneumatického chapadla. Jelikož je součástkou potřeba otočit o 360° jsou trny poháněny krokovým motorem. Jelikož po upnutí součástka leží na lyžinách a nebylo by jí možno otočit, jsou vedení v jedné části pohyblivé v ose y. Pohyb bude realizovat pneumatický válec. Klapka, která zastaví součástku v dané pozici, bude také poháněna pneumatickým válcem. Po provedení optické kontroly je součástka v závislosti na výsledku odděleny vadné kusy od bezvadných. Toto třídění je možno provést několika způsoby. Poměrně elegantní řešení je pomocí tlakového vzduchu. Proud vzduchu bude realizovat tryska. Informaci o tom zda je součástka před tryskou poskytne optické čidlo. To zda bude součástka

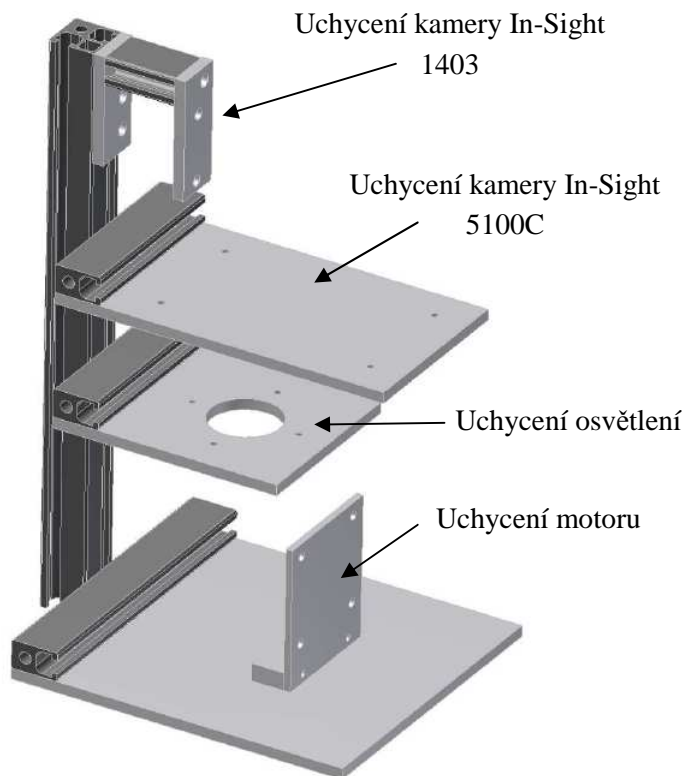
odstraněna nebo ne rozhodne řídicí systém v závislosti na výsledku analýzy. Zkontrolované součástky jsou před trysku dopraveny pásovým dopravníkem. Na konci tohoto dopravníku bude operátor skládat bezvadné součástky do připravených přepravek. Mechanika kontrolního stanoviště je zobrazena na obr. 16. Jedná se o návrh možného řešení, který bohužel není realizován z důvodu nezájmu zadavatele. V následujících odstavcích bude popsána simulace pohonů jednotlivých funkčních částí spolu s řídicím systémem a analýzou obrazu realizovanou v kameře. Pro ověření funkce musel být vyroben přípravek, pomocí něhož bude možné simulovat pohyb před kamerou.



Obr. 16 Mechanika kontrolního stanoviště

6.1 Přípravek pro polohování

Proto, aby bylo možno otestovat vlastnosti a vhodnost volby již zmíněné inteligentní kamery, jsem vytvořil jednoduchý přípravek (obr. 17). Jedná se o polohovatelné upevňovací plochy pro kameru a osvětlení, tak aby bylo možno dosáhnout vhodné konfigurace. Analyzovaný objekt je před kamerou polohován pomocí krokového motoru a to v případě hledání vad na jeho vnějších plochách. Pokud jsou měřeny rozměry, je objekt umístěn staticky před kamerou. Krokový motor je řízen mikrokontrolerem Freescale MC908QY4CPE. Toto zařízení slouží pouze k ověření a ladění funkce programu kamery. Výsledné kontrolní stanoviště by mělo řídit PLC. Tento způsob řízení byl zvolen pro svou relativní jednoduchost a také proto, že nebylo k dispozici vhodné PLC s budičem krokového motoru. Motor vykonává pouze stále se opakující rotační pohyb o velikosti jednoho kroku za 90ms, takže je toto řešení zatím postačující. Kamerou a osvětlením lze pohybovat v a kolem osy y.



Obr. 17 Přípravek

6.1.1 Řídící deska krokového motoru

Jelikož je nutné otočit součástkou v prostoru kamery o 360° navrhnul jsem řídící desku, která budí cívky krokového motoru a generuje spouštěcí signál pro kameru. Vzhledem k typu úlohy jsem jako prvek generující tyto signály použil mikrokontrolér firmy Freescale MC908QY4CPE. Tento mikrokontrolér je doplněn o výkonové a ochranné obvody. Jelikož buzený krokový motor je čtyřfázový, je deska vybavena čtyřmi výkonovými výstupy. Pro možnost ovládání otáčení obsahuje diskretní vstup. Dále diskretní výstup pro spouštění kamery. Pro mé účely je toto řešení dostačující, pokud by došlo k realizaci v průmyslu byl by zřejmě použit značkový budič krokového motoru. Účelem je pouze prověřit chování kontrolního řetězce za dynamických podmínek a simulace reálného řešení.

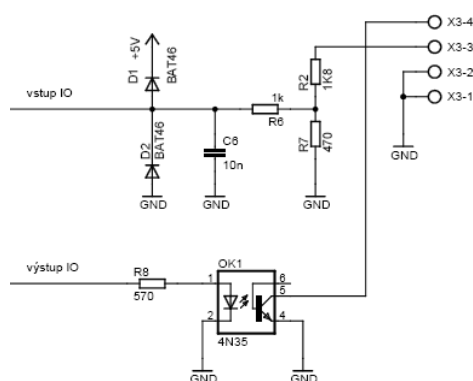
Jako budiče výkonových výstupů jsem zvolil MOSFET tranzistory IFR540. Tyto poskytují dostatečnou výkonovou rezervu pro buzení cívek motoru. Výstupy jsou vybaveny ochranou diodou zapojenou mezi kolektor a emitor tranzistoru. Již samotné tranzistory ve své vnitřní struktuře mají diodu začleněnou, avšak pro zvýšení ochrany jsou doplněny navíc externí rychlou shottkyho diodou. Dioda chrání spínací prvky před vysokým indukovaným napětím při zániku pole. Jelikož se tranzistory typu MOSFET vyznačují vysokou vstupní kapacitou řídící elektrody, je nutné zařadit mezi ně a mikrokontrolér rezistor. Důvodem je proudová špička způsobená nabíjením této kapacity, která by mohla poškodit výstup mikrokontroléru. Použitý tranzistor má vstupní kapacitu 1960pF. Hodnotu předřadného rezistoru jsem zvolil 10kΩ.

Diskretní výstup je oddělen od mikrokontroléru optočlenem. Diskretní vstup tvoří dolní propust, kombinace ochranných diod vstupu mikrokontroléru a napěťové přizpůsobení TTL logice pomocí děliče napětí.

Mimo jiné je deska vybavena dvěma stabilizátory napětí pro dvě výkonové úrovně. Jeden slouží k napájení mikrokontroleru a druhý krokového motoru.

6.1.1.1 Návrh ochran vstupu a výstupu

Dělič diskretního vstupu má za úkol snížit hodnotu napětí dvaceti čtyř voltové logiky PLC na logiku TTL, s níž pracuje mikrokontrolér. Proud děličem jsem zvolil $10mA$. podle Ohmova zákona jsou hodnoty odporů $R2 = 1.8k\Omega$ a $R7 = 470\Omega$. Filtr případných zákmitů je standardní RC dolní propust prvního řádu. Mezní kmitočet jsem zvolil $100kHz$. Z čehož plyne, že hodnota odporu je $R6 = 1k\Omega$ kondenzátoru $C6 = 10nF$. K ochraně před přepětím na vstupu jsou použity rychlé shottkyho diody $D1$ a $D2$ BAT46. Jejich zapojení je vidět ve schématu v příloze. Výstup je oddělen pomocí optočlenu 4N35. Proud a napětí na vstupní diodě jsem podle dokumentace zvolil $10mA$ a $1,7V$. Podle Ohmova zákona je potřebný předřadný odpor roven hodnotě $R9 = 330\Omega$.



Obr. 18 Ochranné obvody

6.1.1.2 Program mikrokontroléru MC68HC908QY4CP

Jak již bylo popsáno výše, použil jsem mikrokontrolér firmy Freescale. Program byl napsán v prostředí Code Warrior jazykem C. Pro usnadnění tvorby kódu a také vzhledem k tomu, že návrh této desky nebyl hlavním bodem mé diplomové práce, jsem využil dodávaného nástroje ProcessorExpert.

Hlavní funkcí aplikace je postupně nastavovat čtyři výstupy v přesně definovaných intervalech danou binární kombinací. Interval je dán časovačem $T1$, který každých $90ms$ vyvolá přerušení, v němž se zapíše binární kombinace na čtyři výstupy *portu B*. Současně se aktivuje druhý časovač $T2$, který slouží k sejmutí obrazu a provedení analýzy kamerou. V přerušení tohoto časovače je nastaven bit číslo pět *potru B*. Tento bit je připojen k trigrovacímu vstupu kamery. Při jednom přerušení od časovače $T1$ dojde k otočení o jeden krok a následně ke spuštění programu v kameře. Toto se neustále opakuje, dokud je bit 3 *portu A* nastaven na hodnotu jedna. Tento vstup slouží k aktivaci provedení kontroly, při příchodu nové součástky do prostoru před kamerou. V případě je-li bit 3 *portu A* nastaven na hodnotu 0 dochází pouze k trigrování kamery, která vyhodnocuje, zda před ní není součástka. Pokud přijde před kameru nová součástka, nastaví PLC bit 3 *portu A* což způsobí zahájení nové analýzy.

Kód programu přerušení časovačů $T1$ a $T2$:

```
void T1_OnInterrupt(void)
{
    PTB_PTB4=0;           //nastavení trigrovacího výstupu na 0
    otoc=PTA_PTA3;         //kontrola bitu 3 portu A
}
```

```

    if(otoc==1){
        i++;
        if(i==1){
            PTB=0b00000001;
            TI2_Init();
        }
        if(i==2){
            PTB=0b00000010;
            TI2_Init();
        }
        if(i==3){
            PTB=0b00000100;
            TI2_Init();
        }
        if(i==4){
            PTB=0b00001000;
            TI2_Init();
            i=0;
        }
    } else{
        //v případě PTA_PTA3=0 pouze trigrování kamery
        TI2_Init();
    }
}

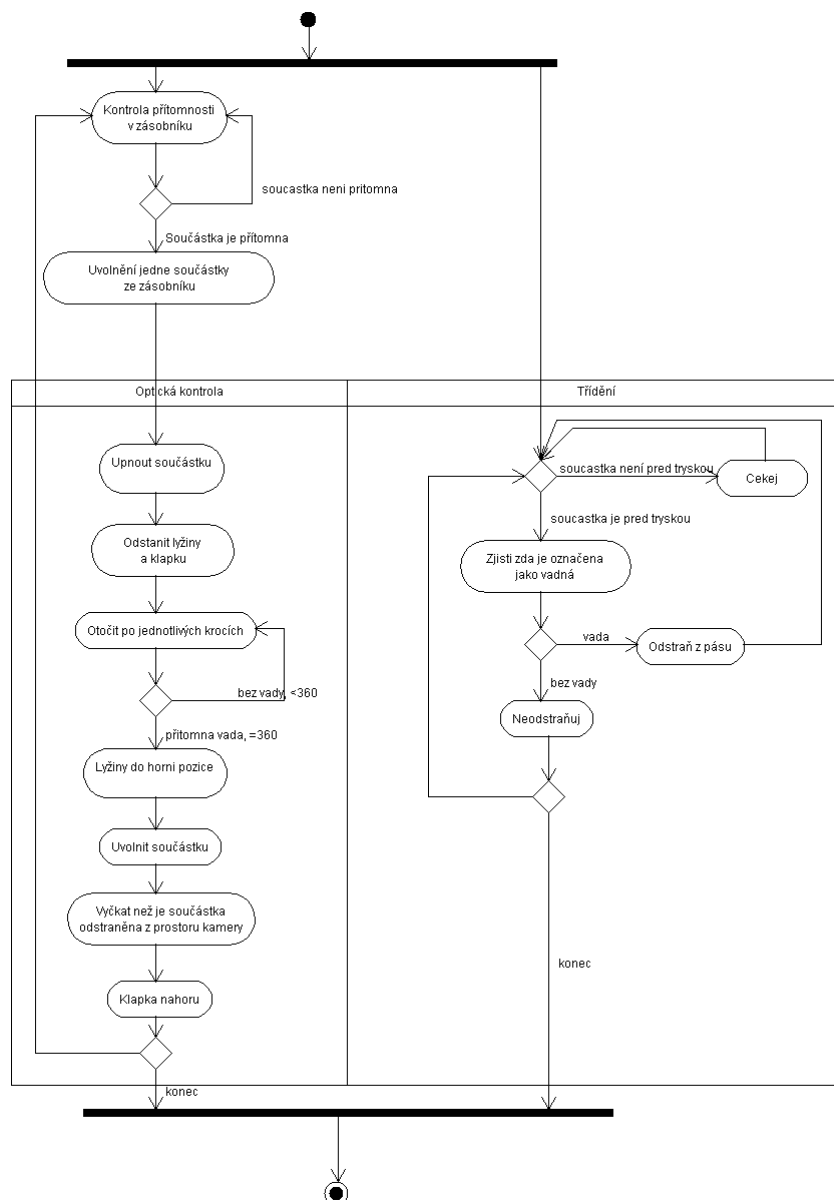
void TI2_OnInterrupt(void)
{
    PTB=0b00010000;
    TI2_Disable();
}

```

6.2 Řídicí systém polohovacího mechanismu

Úkolem polohovacího mechanismu (manipulátoru) je dopravit součástku do pozice, ve které je prováděna kontrola. Součástky jsou umístěny v zásobníku a pomocí podavače jsou postupně dávkovány do pozice kamery. Dále je součástka upnuta do definované pozice pomocí pneumatického chapadla. Jelikož má tvar válce a kontrola musí proběhnout po celém obvodu, je jí otočeno o 360 stupňů. To zajišťuje krokový motor připevněný na trny upínacího prvku. Po úspěšně provedené kontrole uvolní chapadlo sevření a součástka dopadne na dopravníkový pás. Třídění součástek, které neprošly kontrolou, zajišťuje vzduchová tryska, tak že je odstraní z pásu do připraveného zásobníku. Bezvadné součástky přemísťuje operátorka do připravených boxů.

Činnost manipulátoru řídí OPLC Unitronics Vision 120. Jedná se o tzv. OPLC tedy programovatelný automat doplněný o display a klávesnici. Což rozšiřuje možnosti samotného PLC o vytvoření rozhraní člověk stroj (HMI). Toto OPLC disponuje šestnácti digitálními výstupy, z nichž dva jsou konfigurovatelné jako vysoko rychlostní (HSO) a dvacet dvěma digitálními vstupy, z nichž jsou čtyři konfigurovatelné jako vysokorychlostní čítače (HSC) s kanály A, B. Programovacím jazykem je klasický LAD diagram.



Obr. 19 Diagram aktivit řídicího systému

Vzhledem k tomu, že k samotné realizaci nedojde, byl celý systém manipulátoru simulován pomocí přípravku Festo Didactic. Jedná se o školící přípravek se sadou pneumatických válců a ventilů. Důvodem simulace byla představa reálné funkce systému s pneumatickými prvky. Pro pohon jednotlivých částí byly zvoleny dvojčinné a jednočinné pneumatické válce vybavené snímačem koncové polohy. Chapadla byla simulována sériově spojenými dvojčinnými válci. Všechny pneumatické prvky jsou ovládány elektromagnetickými ventily.

Pracovní cyklus kontrolního zařízení je následující. Pokud je v zásobníku minimálně jedna součástka je pomocí podavače uvolněna na dopravníkové vedení, pomocí něhož se samospádem dostane před objektiv kamery. Zde je upnuta pomocí pneumatického chapadla vybaveného otočnými trny s krokovým motorem. Po upnutí dojde k nastavení dopravníkové lavice a jistící klapky do dolní polohy, tak aby bylo možnou součástkou otočit o 360°. Nyní krokový motor otočí součástkou o 360° nebo dokud není kamerou zaznamenána vada na jejím povrchu. Po úspěšně provedené kontrole je dopravníková lavice nastavena do horní polohy a součástka je uvolněna ze sevření chapadel. Ta jakmile opustí prostor kamery, dopadá na dopravník. Klapka je nastavena do horní pozice. Jakmile je

klapka nahoře je možné dopravit ke kontrole další součástku. O oddělení vadných kusů se stará vzduchová tryska, která tento kus pomocí tlakového vzduchu odfoukne do připraveného boxu. Jelikož nebyla možnost zakoupit případně půjčit potřebné měniče a budiče pro ovládání pohonu podavače, dopravníku a otočného trnu chapadla je signalizace spuštění pohonu diskretním výstupem. Aby bylo možno ověřit funkci detekce, navrhnul jsem budič krokového motoru, který otáčí trny chapadla, v nichž je upnuta součástka. Jeho funkce je popsána v předchozí kapitole. Jak jsem již napsal výše, program byl vytvořen na OPLC Unitronics Vision 120. V následující tabulce je uveden přehled vstupů a výstupů.

Tabulka 2: Přehled vstupů/výstupů PLC

	Název	Adresa	Popis
1.	opticke_cidlo1	I0.0	čidlo přítomnosti součástky v zásobníku
2.	soucastka_pred_kamerou	I0.1	výstup z kamery signalizující přítomnost před kamerou
3.	zkontrolovano	I0.2	výstup z kamery kontrola ukončena
4.	pred_tryskou	I0.3	optické čidlo přítomnost součástky před tryskou
5.	soucastka_ko	I0.4	výstup z kamery signalizující vadu na součástce
6.	koncak_chapadlo	I0.5	koncový snímač pneumatického chapadla
7.	koncak_lavice	I0.5	koncový snímač pneumatického pohonu lavice
8.	koncak_klapka	I0.6	koncový snímač pneumatického pohonu klapky
9.	krovy_motor1	O0.0	pohon podavače
10.	ventil1_valec1_lavice	O0.1	pohon lavice
11.	ventil2_valec2_klapka	O0.2	pohon klapky
12.	ventil3_chapadlo	O0.3	pohon chapadla
13.	ventil4_tryska	O0.4	tryska
14.	krokovy_motor2	O0.5	pohon zajišťující otočení součástky o 360°
15.	krokovy_motor3	O0.6	pohon dopravníku

6.2.1 Registrace vadných součástek

pokud je součástka označena za vadnou
pak se zapíše na řádek číslo čítač
zkontrolováno hodnota čítač
zkontrolováno

pokud je součástka před tryskou
je hodnota z řádku číslo čítač *před*
tryskou zapsána do proměnné
registr

Sloupec 1
0
1
0
3
0

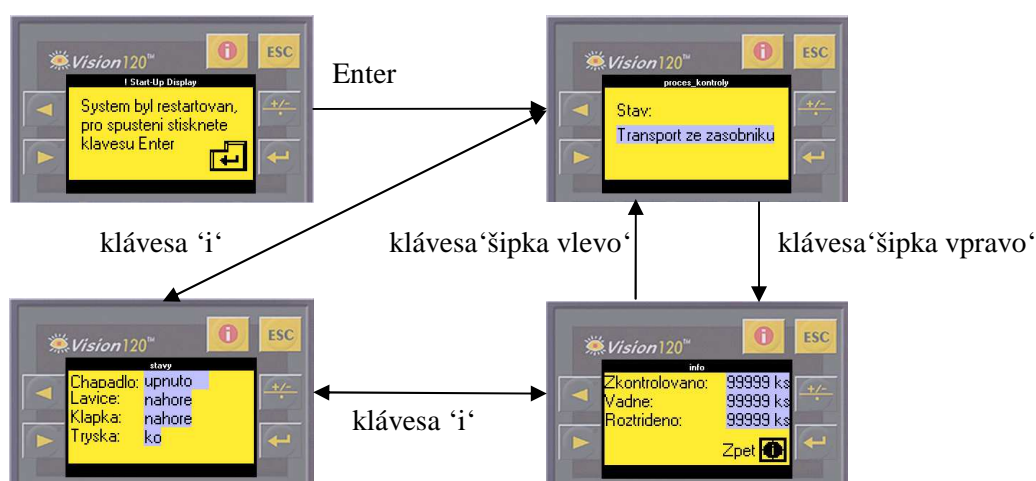
pokud je hodnota proměnné *registr* rovna
proměnné čítač *před tryskou* je součástka
odstraněna z pásu

Obr. 20 Zápis do datové tabulky

Jelikož součástky před třídící tryskou přichází v určitém pořadí ne však ve stejných časových intervalech, musí si řídicí systém pamatovat, která z nich byla označena jako vadná. Toto zajišťuje datová tabulka, v níž se postupně zaznamenává pořadí součástky a je-li součástka před tryskou, řídicí systém provede kontrolu, zda není označena jako vadná. Na následujícím obrázku je uvedeno, jakým způsobem je zapisováno do datové tabulky.

6.2.2 Rozhraní člověk - stroj (HMI)

Vzhledem k tomu, použité PLC je vybaveno displejem a klávesnicí, vytvořil jsem jednoduchou vizualizaci kontroly. Po prvním spuštění nebo restartu je obsluha vyzvána ke stisku klávesy enter. V této fázi dojde k inicializaci pohonů a nastavení systému do počátečního stavu. Na displeji je zobrazena obrazovka se stavy kontrolního stanoviště. Pomocí klávesy ,i' je možno přepínat mezi obrazovkou *info*, kde je zobrazen stav důležitých pohonů. Klávesou šipka vpravo a vlevo lze zobrazit počet zkontrolovaných, vadných a rozříděných součástek.



Obr. 21 Obrazovky vizualizace

6.3 Program In-Sight Explorer

Spolu s kamerou je dodáván program In-Sight Explorer určený ke tvorbě programu analýzy obrazu. Obsahuje poměrně bohatou knihovnu funkcí, které se vkládají do plochy připomínající tabulkový procesor Excel, od kterého se syntaxe programování nijak neliší. [2]

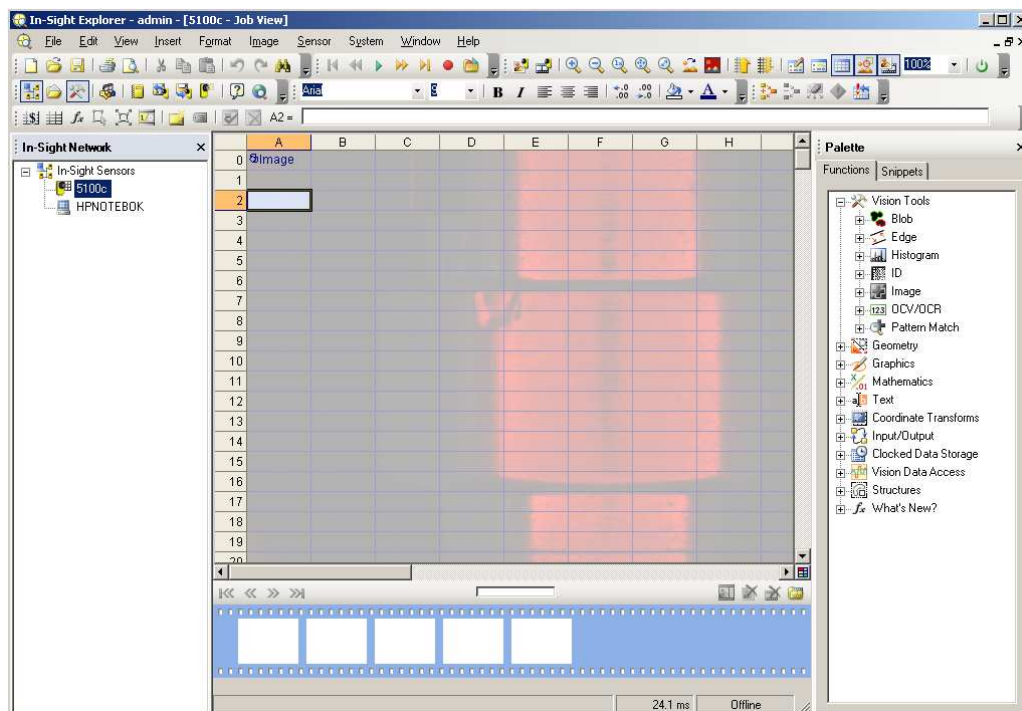
Vpravo je vidět paleta funkcí, uprostřed programovací tabulka a vlevo jsou zobrazeny připojené kamery v síti Ethernet. Program umožňuje také sledovat průběh analýzy a zpracování za běhu na hardwaru kamery.

K velmi zajímavým funkcím patří hledání přednastaveného vzoru *Pattern Match*, *Blob* analýza, hledání hran (přímek, kruhů, křivek), čtení čárového a 2D kódu, funkce ORC, operace s obrazem (filtrace, matematické operace). Mimo tyto funkce zpracovávající obraz, disponuje program také funkcemi pro obsluhu externích vstupů a výstupů, matematickými operace (statistika, trigonometrické funkce), obrazovými transformacemi.

Vytvořený kód lze uložit jako tzv. *Snippet* a ten potom využívat kdekoliv v programu jako svou vlastní funkci.

Průběh programu lze v online režimu sledovat v reálném čase na monitoru počítače. Funkci analýzy lze ukládat na pevný disk počítače a následně přehrát k pozdějšímu rozboru.

V síti Ethernet lze komunikovat také s PLC k tomu je určena speciální komunikační knihovna s GSD soubory.



Obr. 22 Okno programu In-Sight Explorer

6.4 Detekce vad

6.4.1 Detekce hran

Hrana představuje oblast, ve které se náhle mění hodnota jasu. Hrana v obraze je dána vlastnostmi obrazového elementu a jeho okolí. Je určena tím, jak náhle se mění hodnota obrazové funkce $f(x,y)$. Matematickým nástrojem pro studium změn funkce dvou proměnných jsou parciální derivace. Změnu funkce udává její gradient, vektorová veličina ∇ , určující směr největšího růstu funkce (směr gradientu) a strmost tohoto růstu (velikost, modul gradientu). Pixely s velkým modulem gradientu se nazývají hranami. Pro spojitou obrazovou funkci $f(x,y)$ jsou velikost gradientu $|\nabla f(x,y)|$ a směr gradientu ψ dány vztahy:

$$|\nabla f(x,y)| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad (8)$$

$$\psi = \arg\left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)$$

kde $\arg(x,y)$ je úhel (v radiánech) mezi souřadnou osu x a radiusvektorem k bodu o souřadnicích (x,y) . V digitalizovaném obraze aproximujeme parciální derivace diferencemi:

$$\Delta_x g(x,y) = g(x,y) - g(x-n,y)$$

$$\Delta_y g(x,y) = g(x,y) - g(x,y-n) \quad (9)$$

kde n je celé číslo.

Pro nalezení hran se většinou využívají jednoduché operátory aproximující gradient. To se provádí pomocí konvolučních masek aproximující derivace obrazové funkce. Diskrétní konvoluce s jádrem h je dána vztahem:

$$g(x, y) = \sum_{(m,n) \in O} h(x-m, y-n) f(m, n) \quad (10)$$

Kde $f(m, n)$ je vstupní obraz a $h(m, n)$ konvoluční maska. Přístup je založen na představě, že se celý obraz systematicky (např. po řádcích) prochází. Kolem reprezentativního bodu je zkoumáno malé okolí O , často malý obdélník. Výsledek analýzy je zapsán do výstupního obrazu jako hodnota reprezentativního pixelu.

Nejběžnějšími typy konvolučních masek jsou Robertsův, Laplaceův, Sobelův, Robinsonův operátor.

Příklad Sobelova operátoru pro detekci horizontálních hran: [1]

$$h = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

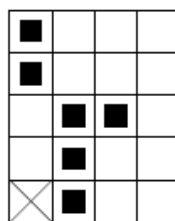
6.4.2 Morfologické operace

Metody matematické morfologie představují prostřední část v posloupnosti zpracování obrazu. Výsledek je reprezentován binárním obrazem. Metody matematické morfologie se používají především pro:

- předzpracování (odstranění šumu, zjednodušení tvarů objektů)
- zdůraznění struktury objektu (kostra, ztenčování, zesilování, výpočet konvexního obalu, označování objektů)
- Popis objektů číselnými charakteristikami (plocha, obvod, projekce, atd.)

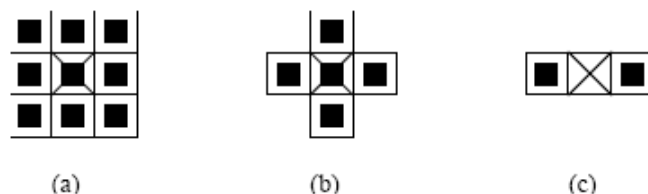
Matematická morfologie vychází z vlastností bodových množin. Lze zpracovávat jak obrazy s více úrovněmi jasu, tak i binárními obrazy.

Binární obraz je reprezentován bodovou množinou. V digitálním zpracování obrazu se používá diskrétní vyjádření bodových množin. Body příslušející objektům v rovině jsou popsány dvojicí svých celočíselných souřadnic. Body doplňku bodové množiny jsou označeny 0. Počátek souřadnic (0,0). Na obr. 13 je ukázka bodové množiny, je počátek označen křížkem.



Obr. 23 Příklad bodové množiny

Prakticky používané morfologické transformace jsou realizovány jako relace obrazu (bodové množiny X) s jinou menší bodovou množinou B , které se říká strukturní element.



Obr. 24 Příklady strukturních elementů

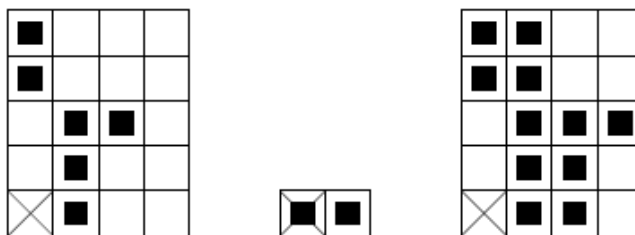
Všimněme si, že počátek nemusí být bodem strukturního elementu, viz obr. 14c. Elementům, které mají stejné vlastnosti pro různé směry, říkáme izotropické. Morfologickou transformaci si lze představit, jako bychom strukturním elementem B systematicky pohybovali po obraze. Bodu obrazu, který se shoduje s počátkem souřadnic strukturního elementu, říkáme okamžitý bod. Výsledek relace mezi obrazem a strukturním elementem (může nabývat hodnoty 0 nebo 1) je zapsán do okamžitého bodu obrazu.

Základními transformacemi matematické morfologie jsou dilatace, eroze, otevření a uzavření. [1]

6.4.2.1 Dilatace

Dilatace skládá body dvou množin pomocí vektorového součtu. [1]

$$X \oplus B = \{d \in E^2 : d = x + b, x \in X, b \in B\} \quad (11)$$



Obr. 25 Příklad dilatace

6.4.3 Segmentace prahováním

Nejjednodušším segmentačním postupem je prahování. Vychází ze skutečnosti, že mnoho objektů či oblastí obrazu je charakterizováno konstantní odrazivostí či pohltivostí svého povrchu. Pak je možné využít určené jasové konstanty prahu k oddělení objektu od pozadí. Vzhledem k nenáročnosti je nejrychlejší segmentační metodou, lze ji provádět v reálném čase.

Prahování je transformace vstupního obrazu f na výstupní (segmentovaný) binární obraz g podle vztahu

$$g(i, j) = \begin{cases} 1 & \text{pro } f(i, j) \geq T \\ 0 & \text{pro } f(i, j) < T \end{cases} \quad (12)$$

kde T je předem určená konstanta nazvaná práh a $g(i, j) = 1$ pro obrazové elementy náležející po segmentaci objektům a $g(i, j) = 0$ pro elementy pozadí (nebo naopak).

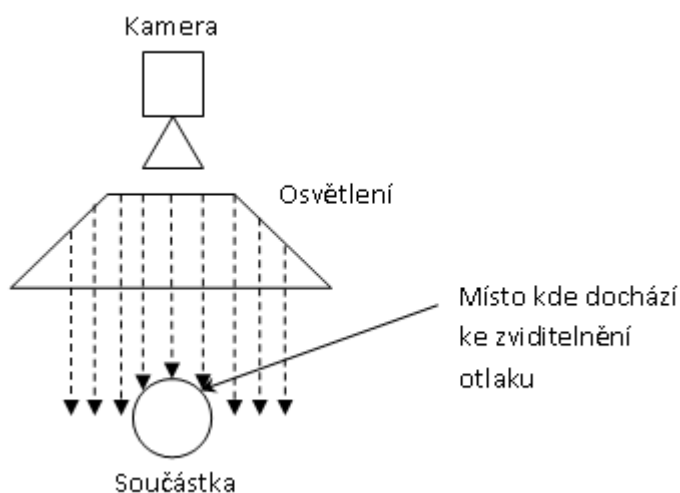
Správná volba prahu je pro úspěšný výsledek prahování zásadní. Hodnotu prahu lze určovat interaktivně nebo pomocí některé z metod automatického určování prahu. Jen málokdy je možné úspěšně prahovat s tímž prahem v celé ploše obrazu, dokonce ani ve zcela jednoduchých scénách nemusí dát globální prahování s jediným prahem požadované segmentační výsledky. To je způsobeno

změnami jasu objektů i pozadí zaviněnými např. nerovnostmi osvětlení či nestejnými vlastnostmi snímacího zařízení v ploše obrazu. Tehdy může pomoci prahování s proměnným prahem, kdy hodnota prahu je určována podle lokálních vlastností obrazu. Zatímco globální práh je určován z celého obrazu: $T = T(f)$, je lokální práh funkcí polohy: $T = (f, f_c)$. Kde f_c je část obrazu, pro niž je práh určován. Jednou z možností volby části obrazu na podobrazy a určení prahu nezávisle v každém z nich. Pokud v některém z podobrazů postup určování lokálního prahu selže, použijeme práh získaný interpolací sousedních prahů. Každý podobraz prahujeme lokálním prahem.

Jinou modifikací je prahování s více prahy, kdy výsledkem již není binární obraz, ale obraz s velmi omezeným počtem jasových úrovní. [1]

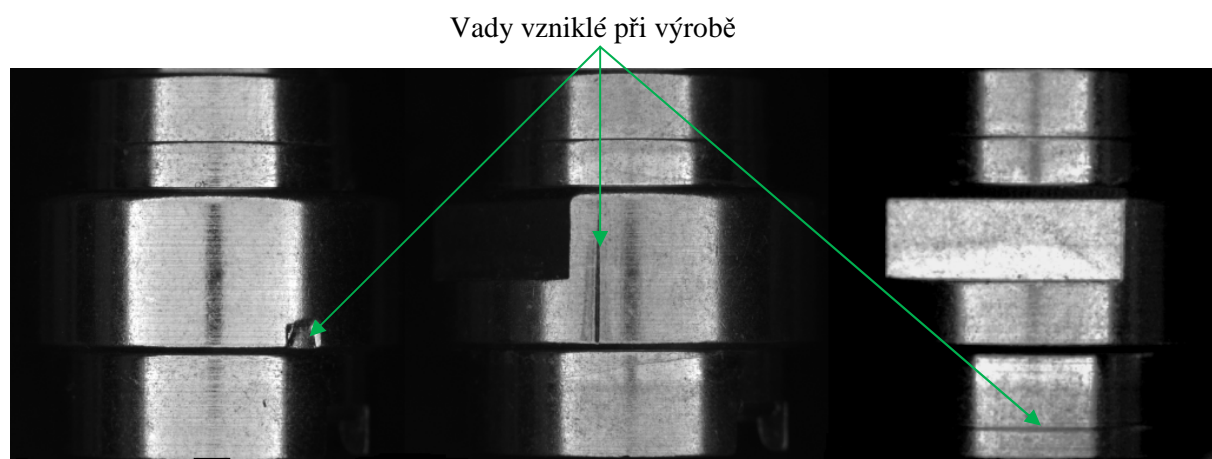
6.4.4 Postup nalezení vad na kontrolované součástce

Vada tvoří porušení jedolité vrstvy povrchu součástky. Vhodným osvětlením dojde k vykreslení struktury vady. Jelikož se jedná o součástku z lesklého kovu, je velmi důležité, aby zdrojem bylo difuzní světlo (rozptýlené světlo). Vlivem zakřivení povrchu součástky dochází ke zviditelnění poruchy povrchu při přechodu světlo stín a naopak. V této poloze hrany vady vrhají stín, který je možné zachytit kamerou.



Obr. 26 Způsob detekce vad

Na povrchu součástky se vyskytuje několik typů vad. Při pohledu z vrchu lze povrch rozdělit na tři oddělené plochy, na kterých se vyskytují odlišné vady. Na následujícím obrázku jsou tyto vady zobrazeny.



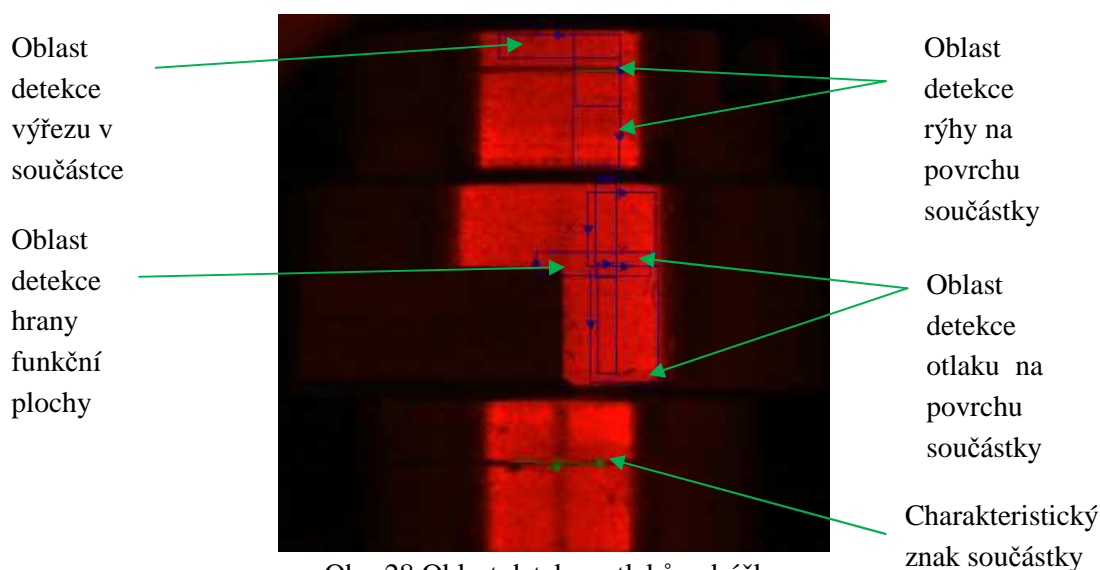
Obr. 27 Vady vyskytující se na povrchu součástky

Pro každý typ vady je nutné zvolit odlišný způsob detekce. V případě otlaku na prostřední ploše je postup detekce následující. Prvním krokem detekce otlaku je porovnání povrchu součástky s referenčním snímkem pomocí nástroje *CompareImage*. Snímek dobrého povrchu součástky je pořízen a uložen v nastavení tohoto nástroje při jeho konfiguraci. Výsledný rozdíl porovnání je převeden na binární obraz užitím funkce *PointFilter* a upraven pomocí dilatace, která se nachází v *NeighborFiltr*. Dilatace je morfologická operace, která skládá body dvou množin pomocí vektorového součtu. Tyto množiny tvoří obraz a strukturální element určitého tvaru a rozměru. Dilatace zvětšuje objekty, používá se k zaplnění malých děr a úzkých zálivů. Hledané vady se po binarizaci vyznačují tím, že netvoří jedolitou plochu, ale obsahují díry. Z toho důvodu je při úpravě obrazu použita dilatace se čtvercovým strukturním elementem 4x4 body.

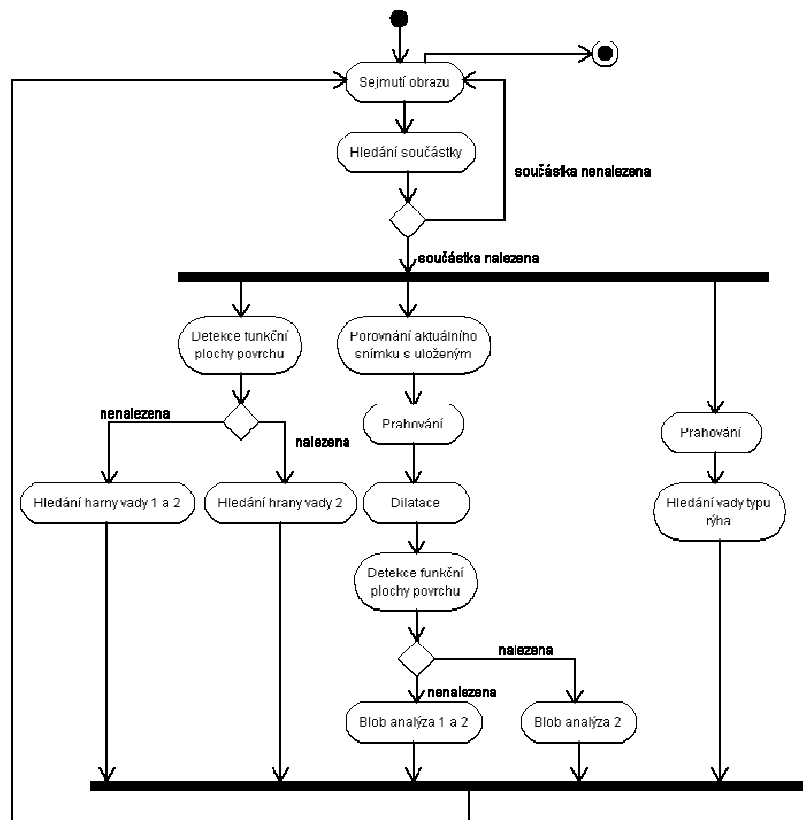
Otlaky jsou poté viditelné jako bílé plochy na černém pozadí. Detekci těchto bílých bodů, které tvoří tzv. kapky, zajišťuje *Blob* analýza představovaná funkcí *ExtractBlobs*. Tato analýza hledá uzavřené objekty stejné barvy v ohraničené oblasti v závislosti na počtu bodů tvořících tyto objekty. Vzhledem k tomu, že se na analyzovaném povrchu vyskytuje funkční plocha, která by způsobovala chybné vyhodnocení, je program doplněn o její detekci pomocí funkce *FindLine*. Tato detekce spočívá v nalezení specifické hrany vyskytující se pouze na dané funkční ploše. Výsledný program využívá dvě *Blob* analýzy a v případě kdy je nalezena výše popsaná hrana, je výsledkem výstup analýzy kde se tato funkční plocha nevyskytuje. Rozhodující podmínkou je velikost *score* analýz a detekce hrany, to zda se jedná o vadnou součástku, rozhodne pravidlo v podmínce *if*.

Mimo tyto tlaky se na povrchu součástky vyskytuje také drážky způsobené vadou na ostří obráběcího nože. Drážka se nachází pouze na jedné z částí povrchů součástky, jak je zobrazeno na obr. 1. Detekce její přítomnosti je založena na hledání hrany v dané oblasti. Vzhledem k tomu, že se na této ploše nachází funkční výřez, je nutné ošetřit jeho přítomnost. Principem je nalezení hrany výřezu a při vyhodnocení přítomnosti drážky je jeho výskyt ošetřen v podmínce *if*.

Poslední vadou je podélný otlak na prostřední ploše. Nalezení této vady je založena na detekci hrany. Jeli, však podélný otlak dostatečně široký, zachytí jej detekce založená na *Blob* analýze používaná v prvním případě. Oblasti analýzy jsou zobrazeny na obr. 28.



Obr. 28 Oblast detekce otlaků a drážky



Obr. 29 Diagram aktivit hledání vad

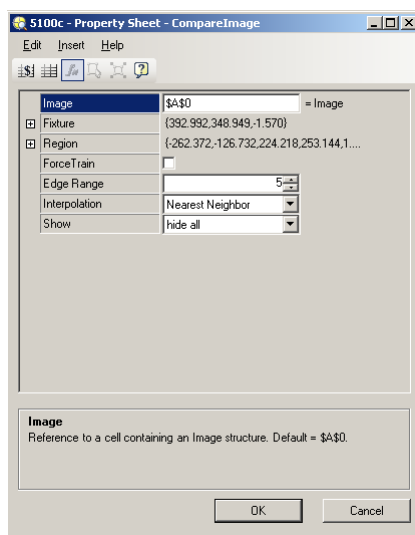
O tom zda je součástka označena jako vadná rozhoduje pravidlo *if*, které je uvedeno níže. To zda buňka nabývá hodnoty 0 nebo 1 závisí na velikosti *Score* jednotlivých nástrojů. Hranice mezi těmito hodnotami je dána nastavením prahu. *Score* označuje úspěšnost nalezení kapky případně hrany. Ke kameře je připojen rozšiřující modul s digitálními vstupy a výstupy. Výsledek detekce vady je signalizován stavem digitálního výstupu *DO0*. Pokud je nalezen otlak, je výstup nastaven na logickou hodnotu 1 jinak nabývá hodnoty 0. Proto, aby byl zkontrolován celý povrch, otočí se součástka o 360 stupňů pomocí krokového motoru. Počet kroků motoru je 59. V každém kroku sejme kamera obraz povrchu a provede výše popsanou analýzu. Spouštění kamery zajišťuje trigrovací vstup na rozšiřujícím modulu. Trigrovací impulzy generuje řídící deska krokového motoru. Jednotlivé snímky jsou čítány a po určitém počtu snímků, jež odpovídá otočení o 360 stupňů je nastaven výstup *DO1*. Tento výstup signalizuje provedení kontroly po celém obvodu součástky. To že je součástka v prostoru kamery signalizuje výstup *DO2*. Ten je nastaven, pokud je v obraze nalezen charakteristický znak součástky, jak je vyobrazeno na obr. 28. Algoritmus detekce vad je zobrazen na obr. 29.

Tvorba programu spočívá v nastavení parametrů jednotlivých funkcí. Při detekci hran je důležité zvolit správně orientaci šipky obdélníku ohraničující oblast detekce hrany. Ta by měla být vždy kolmá k hraně. Dále velikost akceptovatelného prahu a orientaci hrany. Pro nalezení otlaků je použita funkce *ExtractBlob*. Nastavení této funkce spočívá mimo jiné ve vymezení oblasti, ve které dochází k detekci otlaků a volbě maximálního a minimálního počtu bodů reprezentující otlak. Důležité je také zvolit správně barvu pozadí a otlaku. Nastavil jsem tedy barvu pozadí na černou a otlaku bílou, hranice jsou v rozmezí $<100;600>$ bodů. S těmito hodnotami jsem schopen nalézt výše uvedené otlaky vyskytující se na dodaných součástkách.

Pro následnou správnou funkci programu je důležité zachovat nastavení pozice kamery, osvětlení a součástky. Důvodem je pevně nastavená pozice oblastí prováděných analýz a detekcí hran. Vlivem změny výše popsaných nastavení pozic dochází k chybné v detekci vad.

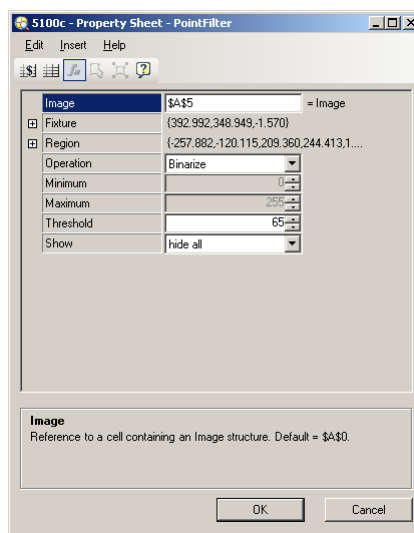
6.4.5 Použité funkce

CompareImage tato funkce slouží k porovnání dvou obrázků. Z nichž jeden je uložen při konfiguraci a druhý je aktuálně sejmutý. Rozdílem je obrázek ve stupních šedi. V poli *Image* je reference na buňku porovnávaného obrázku. *Fixture* slouží k tomu, aby oblast, v které probíhá porovnání, sledovala změny polohy součástky před kamerou. Jelikož se součástka před kamerou pohybuje pouze kolem své osy, není tato možnost použita. Položka *Region* představuje oblast obrázku, ve které probíhá porovnání. Pokud bychom chtěli v průběhu chodu programu měnit uložený obrázek, jež slouží jako referenční pro porovnávání můžeme použít položku *ForceTrain*. *EdgeRange* specifikuje počet pixelů, o který se může shoda lišit. *Interpolation* specifikuje typ interpolace (proložení) pixelů pokud dojde k otočení vzoru v oblasti porovnávání. Lze zvolit *Nearest Neighbor* což znamená, že bude porovnáván nejbližší hodnota pixelu nebo *Bilinear* tedy lineární kombinace čtyř sousedních pixelů. První volba je rychlejší avšak méně přesná. Položka *Show* slouží pouze k tomu zda má být zobrazen výsledek případně oblast ve které porovnávání probíhá.



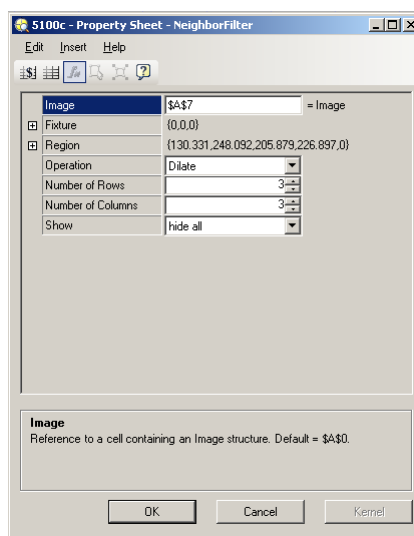
Obr. 30 Funkce *CompareImage*

PointFilter obsahuje několik operací na obrázku ve stupních šedi. První tři pole slouží k stejnému účelu jak bylo napsáno výše. Reference v poli *Image* je výsledek předchozí operace. V mém programu využívám operaci *Binarize*. Jedná se o metodu prahové segmentace, jak bylo popsáno v odstavci 6.4.3. Práh se volí v poli *Threshold*.



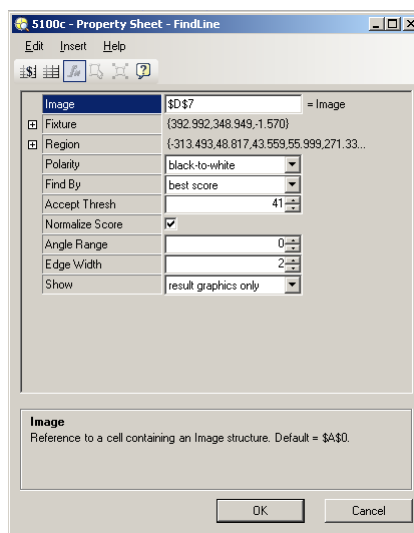
Obr. 31 Funkce *PointFilter*

NeighborFilter funkce obsahuje základní morfologické transformace. V programu používám morfologickou transformaci dilatace. Princip byl popsán výše v odstavci 6.4.2. Typ je transformace je možno vybrat v poli *Operation*. Velikost strukturního elementu je možno zvolit v polích *Number of Row* a *Number of Columns*. V mém případě velikost zvolena 3x3 body.



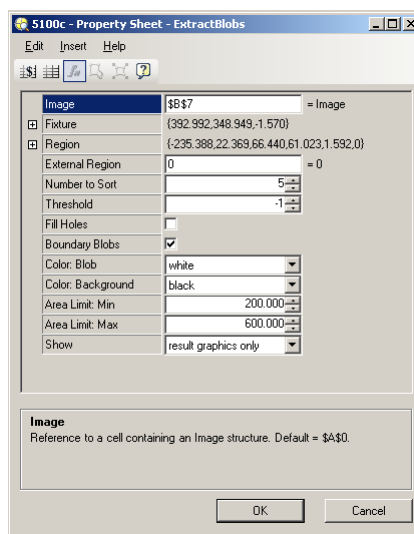
Obr. 32 Funkce *NeighborFilter*

FindLine je funkcí k nalezení hrany tvaru linky. V nastavení této funkce lze zvolit polaritu (*Polarity*) tzn., zda se jedná o hranu na rozmezí černá - bílá nebo bílá – černá nebo obojí. Pole *Find By* slouží k volbě zda je jako výsledek považována hrana s nejlepším výsledkem, první hrana nebo poslední v oblasti analýzy. Hodnotou *AcceptTresh* lze ovlivnit práh od kterého jsou hrany nalézány. Pokud je zaškrtnuto políčko *Normalize Score* je hrubý výsledek nalezené hrany normalizován maximálním kontrastem v oblasti analýzy. Této volby se využívá k nalezení hran v obraze s malým kontrastem. V poli *AngleRange* lze nastavit rozsah úhlu natočení hrany vzhledem k oblasti analýzy ve stupních. *Edge Width* specifikuje jako tloušťku v bodech má hledaná hrana.



Obr. 33 Funkce *FindLine*

Funkce *ExtractBlob* rozdělí analyzovanou oblast na dvě kategorie. Kapka nebo pozadí v závislosti na stupni prahu a velikosti limit. Počet nalezených kapek se nastavuje v poli *Number to Sort*. Hodnota prahu je zadána v poli *Threshold* a limity velikosti kapek v *Area Limit: Min* a *Area Limit: Max*. Barva hledané kapky a pozadí se volí výběrem *Color: Blob* a *Color: Background*. Výběrem *Boundary Blobs* budou do struktury analýzy zahrnuty i ty kapky, které se protínají s hranicí oblasti analýzy. Volbou *Fill Holes* můžeme zahrnout do oblasti kapky případné díry, které se v ní vyskytují.



Obr. 34 Funkce *ExtractBlob*

6.5 Měření rozměrů

6.5.1 Metrické vlastnosti obrazu

Uvažujeme digitální obraz složený z bodů konečných rozměrů uspořádaných do čtvercové vzorkovací mřížky. Některé pojmy, jejichž význam je ve spojitě matematice intuitivně zřejmý nemají pro diskrétní obrazy snadnou analogii. Příkladem je vzdálenost mezi dvěma obrazovými body. Abychom funkci D mohli říkat vzdálenost (metrika) musí mít tyto vlastnosti:

$$\begin{aligned}D(p, q) &\geq 0 \\D(p, q) &= D(q, p) \\D(p, r) &\leq D(p, q) + D(q, r)\end{aligned}\tag{13}$$

V digitálním zpracování obrazu se používá několik definic vzdálenosti. Mějme dva obrazové body se souřadnicemi (x, y) , (h, k) . *Euklidovská vzdálenost* D_E je známa z elementární geometrie a každodenní zkušenosti:

$$D_E((x, y), (h, k)) = \sqrt{(x - h)^2 + (y - k)^2}\tag{14}$$

Její výhodou je názornost. Nevýhodou je složitější výpočet kvůli odmocnině a skutečnost, že výsledkem je neceločíselná hodnota, což neodpovídá kvantovaným hodnotám v digitálních obrazech.

Vzdálenost mezi dvěma body v obraze si lze také představit jako nejmenší počet kroků jednotkové vzdálenosti nutných pro přesun z výchozího do cílového bodu v pravoúhlé diskretizační mřížce. Jsou-li povoleny jen kroky o jedno pole v mřížce buď ve vodorovném, nebo svislém směru získáme vzdálenost D_4 , které se říká *vzdálenost v městských blocích* díky analogii se vzdáleností ve městě s pravoúhlou sítí ulic.

$$D_4((x, y), (h, k)) = |x - h| + |y - k|\tag{15}$$

Když navíc připustíme i jednotkové kroky v diagonálním směru, je odpovídající vzdáleností D_8 , které se říká *šachovnicová vzdálenost*, protože odpovídá nutnému počtu krále na šachovnici:[1]

$$D_8((x, y), (h, k)) = \max\{|x - h|, |y - k|\}\tag{16}$$

6.5.2 Užití smart kamery pro měření rozměrů

Základní otázkou je, samozřejmě, dosažitelná přesnost měření. Pro tvářené díly jsou obvyklé tolerance rozměrů v setinách milimetru, pro obráběné součásti už to často bývají mikrometry. I takového rozlišení lze pomocí kamery dosáhnout, je však třeba znát úskalí použitých měřicích metod.

Parametrem kamery určujícím přesnost měření je rozlišení (počet pixelů) jejího obrazového čipu. Software moderní smart kamery umí navíc využít i tzv. subpixelového rozlišení které umožní stanovit polohu hrany obrazu součástky s až dvacetinásobným rozlišením. Rozlišení $5\mu\text{m}$ (v odpovídajícím zorném poli) je dnes požadavek splnitelný i běžnou smart kamerou s čipem 640×480 pixelů. Pro takové rozlišení potřebujeme samozřejmě získat kvalitní a stabilní obraz měřené součásti. Je však třeba vědět jak objekt nasvítit a sejmout jeho nezkrasený obraz.

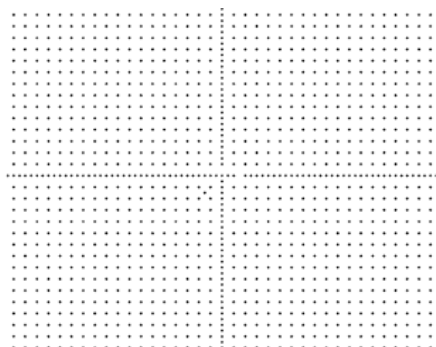
První podmínkou je odpovídající osvětlení. Pro měření rozměrů se nejčastěji používá zadní osvětlení a snímáme siluetu měřené součásti. Osvětlení musí být homogenní a stabilní aby softwarové subpixelové rozlišení bylo vždy prováděno za stejných podmínek. Navíc je třeba správně zvolit barvu i intenzitu světla a eliminovat vliv okolních nekontrolovatelných světelných zdrojů: slunečního světla, osvětlení pracoviště atd. Velký vliv na dosažitelnou přesnost měření má použitá optika. Při měření

prostorových objektů se uplatňuje perspektivní zkreslení – vzdálenější část objektu se jeví menší, než ta, která je kameře blíže. Perspektivní zkreslení koriguje tzv. telecentrický objektiv kterým se úsečka stejného rozměru promítne na čip kamery stejně dlouhá bez ohledu na to, jak je od kamery daleko. Telecentrický objektiv je poměrně nákladný, mimo jiné i proto, že průměr vstupní čočky musí být větší než úhlopříčka zorného pole. U objektivů s velkými čočkami je ovšem obtížnější potlačit ostatní optické vady a proto ne každý dodavatel umí vyrobit telecentrický objektiv vhodný pro měření.

Měří-li se za stále stejných podmínek (je-li například součástka polohována robotem na měřicí stolici), lze se někdy použití nákladné optiky vyhnout. Použije se kalibrační metoda: případné chyby softwarově kompenzujeme srovnáním se součástí považovanou za etalon. Je však třeba věnovat zvýšenou pozornost uspořádání měřicího pracoviště. Perspektivní zkreslení „umožňuje“ vidět i plochy rovnoběžné s osou objektivu. Při použití standardního osvětlení se pak například může stát, že změříme drážku užší než stejně široký výstupek. [9]

6.5.3 Kamery Cognex In-Sight

Důležitou součástí měření rozměrů je kalibrace kamery. Používaná kamera disponuje kalibračním nástrojem, který zavede do obrazu systém souřadnic x, y v reálných jednotkách (např. milimetrech). K tomu slouží vygenerovaná mřížka s body v zadaném intervalu rozestupu těchto bodů. Volba tohoto rozestupu je závislá na rozměrech měřeného objektu a velikosti zorného pole. Tuto kalibrační mřížku je nutné vytisknout v určité přesnosti. Přesnost je dána tiskařským zařízením a ta u klasických tiskáren nedosahuje hodnoty lepší jak $\pm 0,1\text{mm}$. Dalším faktorem ovlivňující výsledek kalibrace je s jakou přesností dodržíme vzdálenost mřížky od čipu kamery. Tato vzdálenost musí být shodná se vzdáleností měřené součástky před kamerou. Vzhledem k přesnosti výroby $\pm 0,1\text{ mm}$ měřené součásti není tento způsob kalibrace příliš vhodný. Příklad kalibrační mřížky je zobrazen na obr. 35.



Obr. 35 Kalibrační mřížka

Lze však zvolit jiný způsob kontroly rozměrů a to tak, že máme jednu součástku o známých rozměrech, kterou prohlásíme za bezvadnou (vyrobenou s danými rozměry) a porovnáním s následně měřenou součástí zjistíme odchylku od rozměrů kalibrační součásti. Pokud tato odchylka nebude vyšší, než přesnost s jakou je součást vyrobena, lze říci, že je kontrolovaná součást vyrobena právě s touto přesností a označit ji jako bez vady.

Celková přesnost měření potom závisí především na rozlišení kamery a přesnosti nástrojů detekce hran. Měl jsem k dispozici dva typy kamer a to s rozlišením 640×480 bodů a 1600×1200 bodů. Vzhledem k tomu, že kamery jsou od stejného výrobce Cognex, přesnost nástrojů detekce u obou těchto kamer je $\frac{1}{4}$ bodu. Přepočet na rozlišení v reálných jednotkách je dán velikostí zorného pole. Následuje výpočet rozlišení a přesnosti pro tyto kamery In-Sight 5100C a In-Sight 1403.

6.5.4 In-Sight 5100C

Jedná se o kameru s rozlišením 640 x 480 bodů. Zvolil jsem objektiv s ohniskovou vzdáleností 25 mm. Vzdálenost od snímané scény je 125 mm. Rozměry zorného pole jsou 23,7 x 17,75 mm.

Rozlišení, kterého lze dosáhnout:

$$\text{rozlišení} = \frac{\text{vzdálenost}_{\text{kratší}} \cdot \text{strany}_{\text{FOV}}}{\text{odpovídající}_{\text{rozlišení}}} = \frac{17,75}{480} = 0,036\text{mm} \quad (17)$$

Přesnost jaké lze dosáhnout je dána přesností nástrojů (ta je ¼ bodu) dodávaného programu a velikostí rozlišení:

$$\text{teoretická}_{\text{presnost}} = 0,25 \cdot 0,036 = 9,24\mu\text{m} \quad (18)$$

6.5.5 In-Sight 1403

Jedná se o kameru s rozlišením 1600 x 1200 bodů. Zvolil jsem objektiv s ohniskovou vzdáleností 25 mm. Vzdálenost od snímané scény je 125 mm. Rozměry zorného pole jsou 21,965 x 16,474 mm.

Rozlišení, kterého lze dosáhnout:

$$\text{rozlišení} = \frac{\text{vzdálenost}_{\text{kratší}} \cdot \text{strany}_{\text{FOV}}}{\text{odpovídající}_{\text{rozlišení}}} = \frac{16,474}{1200} = 0,014\text{mm} \quad (19)$$

Přesnost jaké lze dosáhnout je dána přesností nástrojů (ta je ¼ bodu) dodávaného programu a velikostí rozlišení:

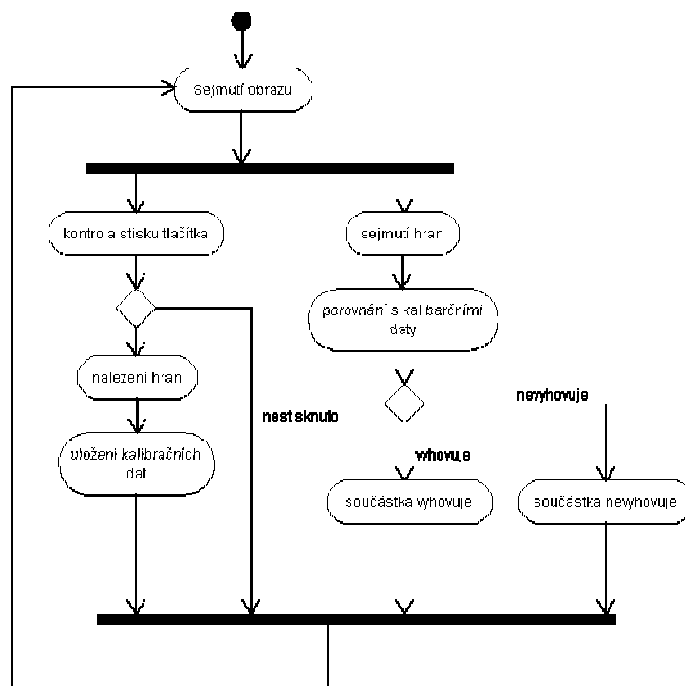
$$\text{teoretická}_{\text{presnost}} = 0,25 \cdot 0,014 = 3,5\mu\text{m} \quad (20)$$

6.5.6 Program pro měření rozměrů

Obrysy a funkční plochy součástky jsou charakteristické svými hranami. Vzdálenost mezi jednotlivými hranami představuje rozměr v počtu bodů. Přepočítání na rozměr ve fyzikálních jednotkách je dáno:

$$\text{vzdálenost} = \text{rozlišení} \cdot \text{pocet}_{\text{bodů}} \quad (21)$$

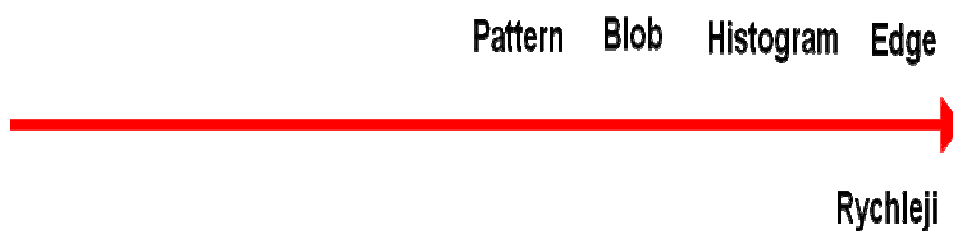
Hrana je oblast v obraze, kde se náhle mění hodnota jasů. Je důležité, aby tato změna byla co možná největší. Toho docílíme vhodným osvětlením a vysokou ostrostí obrazu. Softwarový nástroj In - Sight Explorer obsahuje několik funkcí na detekci hran. V mém programu jsem použil funkci FindSegment. Výstupem je vzdálenost mezi dvěma hranami v počtu bodů. Pro získání kalibračních rozměrů je program doplněn o tlačítko kalibruj. Umístí-li se před kameru kalibrační vzorek, uloží se změřené rozměry. Tato data jsou poté používána jako referenční a slouží k porovnání s právě měřenou součástkou. Pokud je rozdíl vyšší než přesnost výroby, je součástka označena za vadnou. Algoritmus kontroly rozměrů je vidět na obr. 36.



Obr. 36 Diagram aktivit měření rozměrů

6.6 Rychlost kontroly

Vzhledem k tomu, že výroba jedné součástky trvá určitou dobu, je nutné vyčíslit, jak dlouho potrvá kontrola této součástky. Celkový čas je závislý na době transportu součástky před kameru, délce kontroly a době třídění. Čas transportu je neměnný. Udává ho rychlost pohonů. Délka kontroly je proměnná a závisí na tom, zda analyzovaná součástka obsahuje vadu. Dále také na rychlosti, se kterou je kamera schopna provést tuto analýzu. Důležitým faktorem ovlivňujícím rychlost provádění funkcí analýzy je rozlišení kamery, tedy objem dat, na niž jsou operace prováděny a také náročnost funkce. Čas potřebný pro zpracování jednoho snímku pomocí kamery In-Sight 5100C je 85ms. Nejedná-li se o součástku s vadou pak čas potřebný ke kontrole celé součástky je 5,083s. V případě kamery In-Sight 1403 je to 200ms celkem tedy 11,8s. Časová závislost použitých funkcí je znázorněna na obr. 37.



Obr. 37 Závislost rychlosti na použitých funkcích

6.7 Správa uživatelů

Program In-Sight Explorer umožňuje nastavit několik uživatelských profilů. Profil správce (administrátor), který může měnit a konfigurovat chod celého programu. Operátor má právo spouštět program, ovládat jeho chod pomocí tlačítek, otevírat a zavírat nové programy. Monitor může pouze sledovat chod programu.

Další možností, kterou můžeme využít je automatické spuštění vybraného programu po zapnutí napájení. Pokud tedy bude kamera zapnuta načte se nastavený program, poté pokud některý z uživatelů v síti spustí In-Sight Explorer bude vyzván k přihlášení pod jeho uživatelským jménem a heslem.

7 Závěr

Cílem diplomové práce bylo především prověřit možnosti kamerových systémů pro kontrolu povrchových vad obráběných součástí. Většina aplikací kamerových systémů je kontrola čárových kódů, sestavení, obalů, základních rozměrů, atd. Z dosažených výsledků vyplývá, že by bylo možné tento způsob detekce použít, vady však musí dosahovat rozměrů několika stovek pixelů, aby je bylo možno nalézt. Jistým řešením nalezení vad menší velikosti by bylo snímat povrch po menších částech, tedy v menším zorném poli. Důsledkem by však bylo prodloužení doby kontroly a je otázkou zda není v tomto případě vhodnější kontrola manuální.

Abych mohl tuto práci vypracovat, musel jsem se seznámit s problematikou kamerových systémů a především inteligentních kamer. Naučit se s nimi pracovat a pochopit způsob použití dodávaných funkcí pro analýzu obrazu.

Vzhledem k tomu, že se v prvopočátku předpokládala realizace, musel jsem navrhnout mechanickou část samotného kontrolního stanoviště. Z toho důvodu jsem se seznámil s 3D modelováním v programu Inventor. I když k samotné realizaci nedošlo, návrh posloužil k představě při tvorbě řídicího systému. Řídicí systém byl navrhnout na platformě OPLC Unitronics Vision120 se kterým jsem na vysoké škole doposud nepracoval, tudíž jsem se s ním musel taktéž seznámit.

Při návrhu řídicí desky krokového motoru jsem použil mikrokontrolér firmy freescale, čímž jsem využil znalosti získané při studiu.

Kontrolu rozměrů jsem zkoušel také provádět pomocí kalibrační metody popsané v odstavci 6.5.3, ale vzhledem k problémům které jsem již zmínil, nebyla přesnost měření dostatečná. Přistoupil jsem tedy k metodě popsané v další části odstavce 6.5.3. Z výpočtů přesnosti by pro kontrolu rozměrů byla samozřejmě vhodnější kamera In-Sight1403.

Nastavení pozice kamery a osvětlení vzhledem k součástce je velice citlivé na změny, je proto nutné zajistit aby nedošlo ke změně nastavení této pozice. Funkce detekce a měření by neprobíhala správně.

Kontrola rozměrů byla realizována pomocí kamery In-Sight 1403 a detekce vad pomocí In-Sight 5100C.

Získané zkušenosti a poznatky budou dále využity pro rozvoj v této oblasti ve firmě, která tuto práci zadávala.

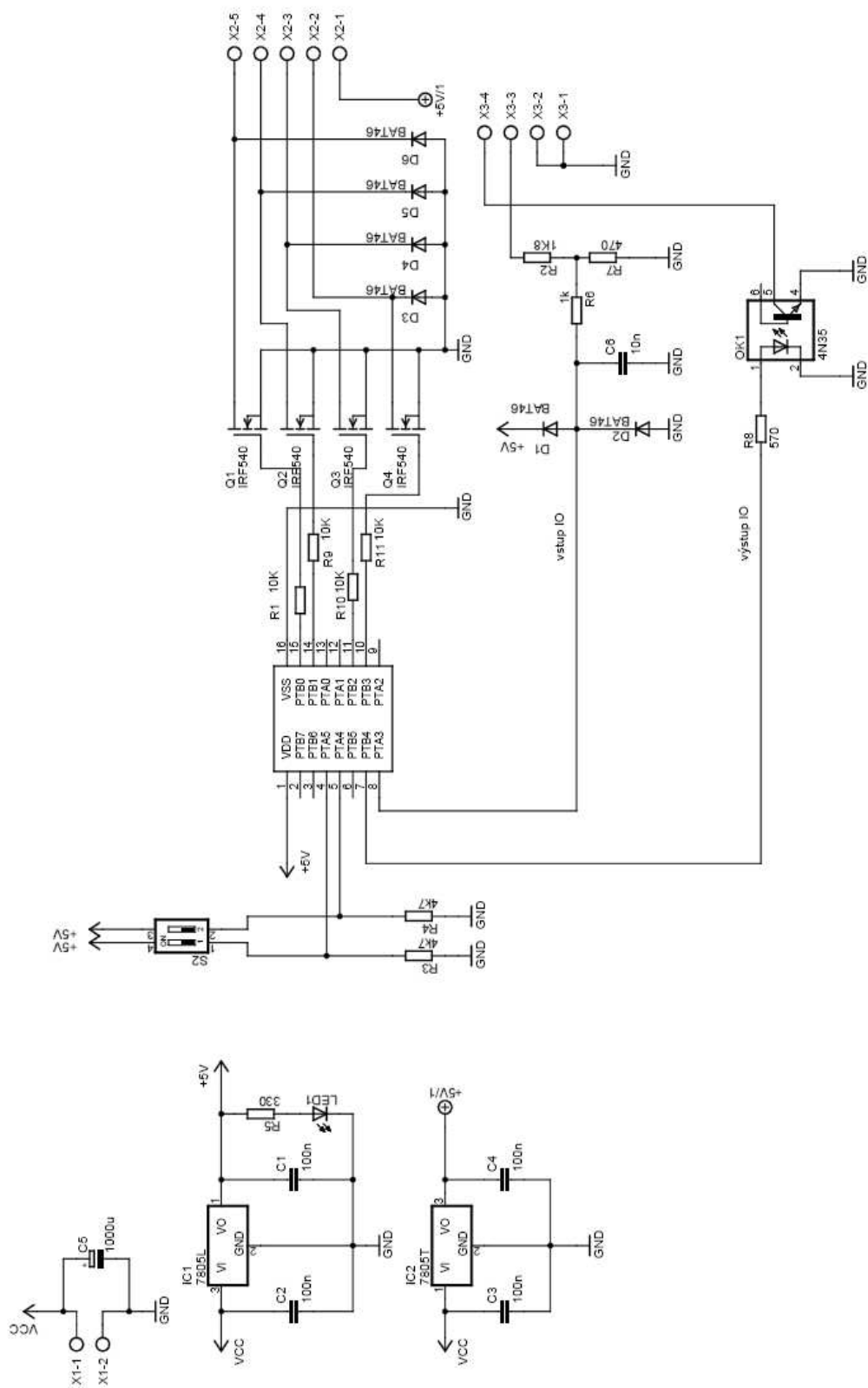
8 Použitá literatura

- [1] Prof.Ing. Václav Hlaváč, Doc.Ing. Miloš Sedláček, Zpracování signálů a obrazu, Vydavatelství ČVUT Praha 2005
- [2] Stránky výrobce kamerových systémů *Cognex*, www.cognex.com
- [3] Stránky výrobce kamerových systémů *Banner*
www.bannerengineering.com/training/div_home.php?div=4
- [4] Článek v časopise *Automatizace*, www.automatizace.cz/article.php?a=1588
- [5] Článek v časopise *Automatizace*, www.automatizace.cz/article.php?a=389
- [6] Stránky firmy *National Instruments*, www.ni.com/vision/
- [7] Stránky firmy *SICK*, www.sick.com
- [8] Stránky firmy *Siemens*, www.siemens.com
- [9] Stránky firmy FCC průmyslové systémy s. r. o
<http://www.fccps.cz/default.asp?inc=inc/articles/kontola-toleranci-rozmeru-strojnich-soucasti-kamerou.htm>
- [10] Srovnal, V.: Regulační systémy. Studijní texty na WWW stránkách fakulty, 2005
- [11] Koziorek, J.: Logické systémy řízení a programovatelné automaty. Sylaby na WWW stránkách katedry, 2002
- [12] Šmejkal L., Martinásková M.: PLC a automatizace. BEN technická literatura, Praha 1999
- [13] Šmejkal L., Martinásková M.: Řízení programovatelnými automaty. Vydavatelství ČVUT, Praha 1998
- [14] Kanisová H., Müller M.: UML srozumitelně. Computer Press, Brno 2004
- [15] Sonka M., Hlavac V., Boyle R.: Image Processing, Analysis, and Machine Vision. Thomson Learning 2008
- [16] Matoušek, D.: Číslicová technika. BEN technická literatura, Praha 2002
- [17] Láníček, R.: Elektronika - obvody, součástky, děje. BEN technická literatura, Praha 1998

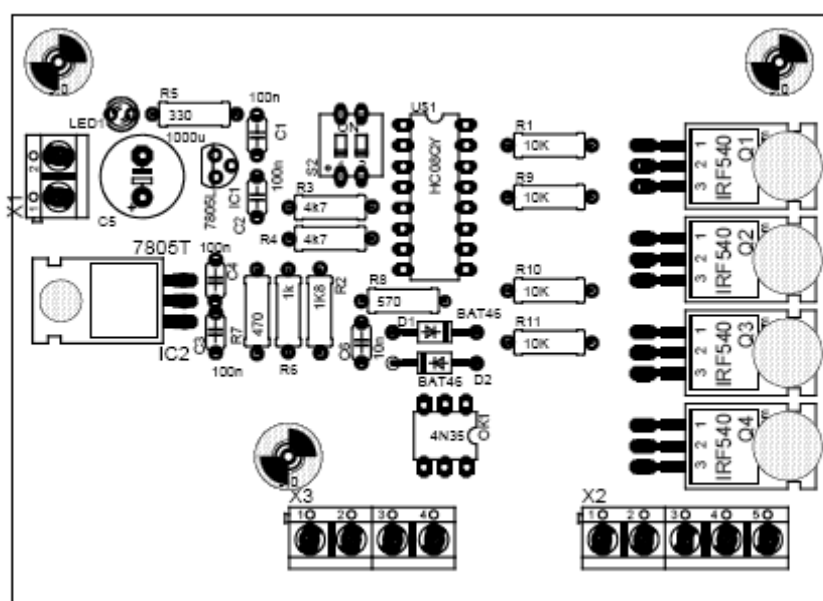
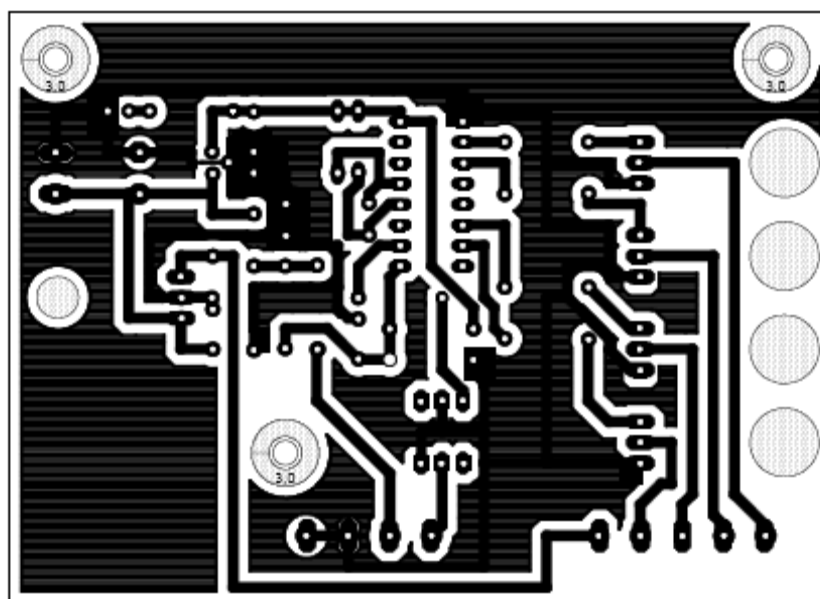
9 Seznam příloh

PŘÍLOHA 1 SCHÉMA ŘÍDÍCÍ DESKY KROKOVÉHO MOTORU	39
PŘÍLOHA 2 DPS ŘÍDÍCÍ DESKY KROKOVÉHO MOTORU	40
PŘÍLOHA 3: UCHYCENÍ KAMERY INSIGHT5100C.....	41
PŘÍLOHA 4 UCHYCENÍ KAMERY INSIGHT1403	42
PŘÍLOHA 5 SIMULACE PNEUMATICKÝCH POHONŮ MANIPULÁTORU	43
PŘÍLOHA 6 SCHÉMA ELEKTROPNEUMATICKÉ ČÁSTI MANIPULÁTORU	44
PŘÍLOHA 7: PROGRAM DETEKCE VAD SOUČÁSTEK	45
PŘÍLOHA 8: PROGRAM KONTROLY ROZMĚRŮ	46

Příloha 1 Schéma řídicí desky krokového motoru



Příloha 2 DPS řídící desky krokového motoru



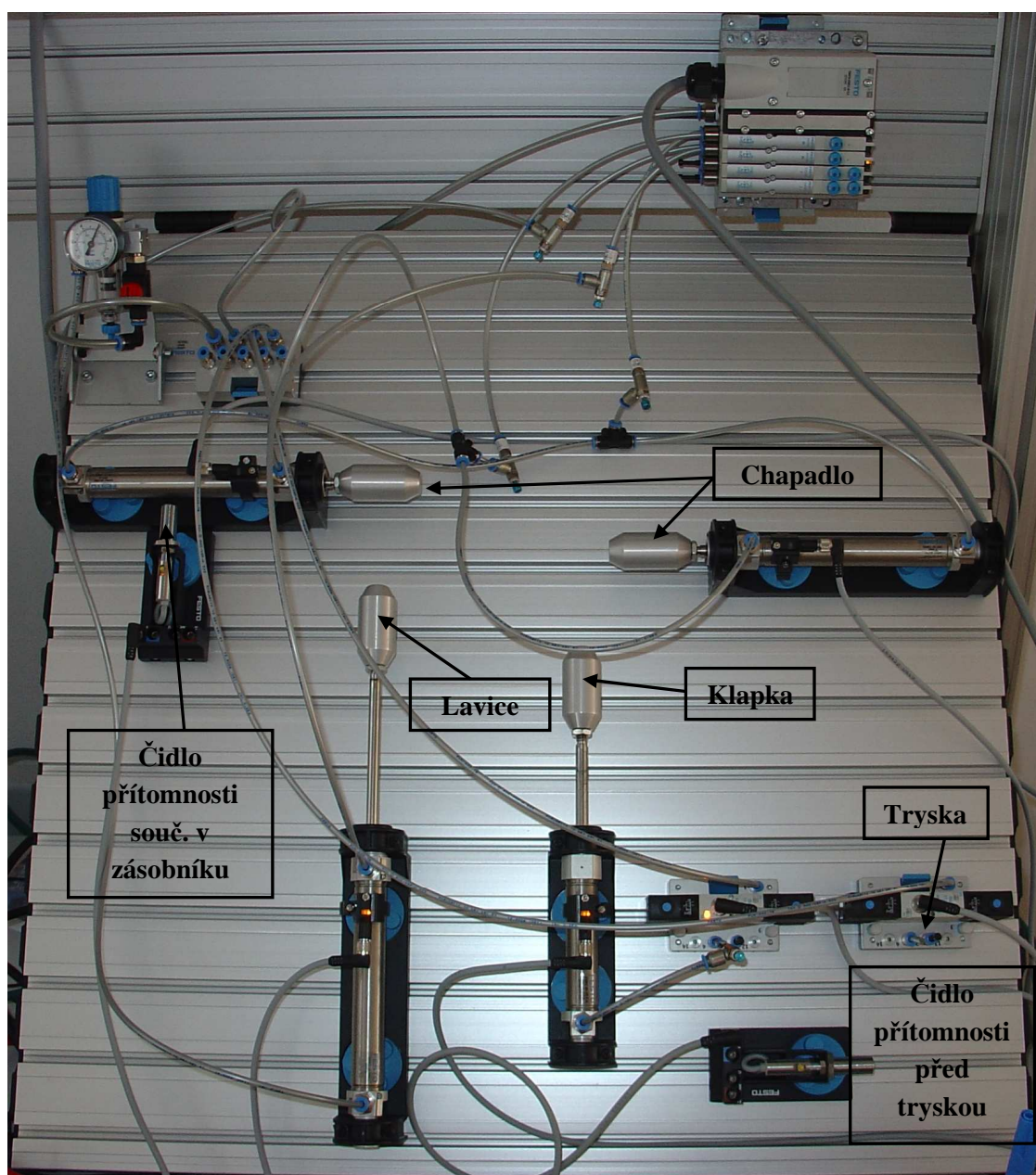
Příloha 3: Uchycení kamery InSight5100C



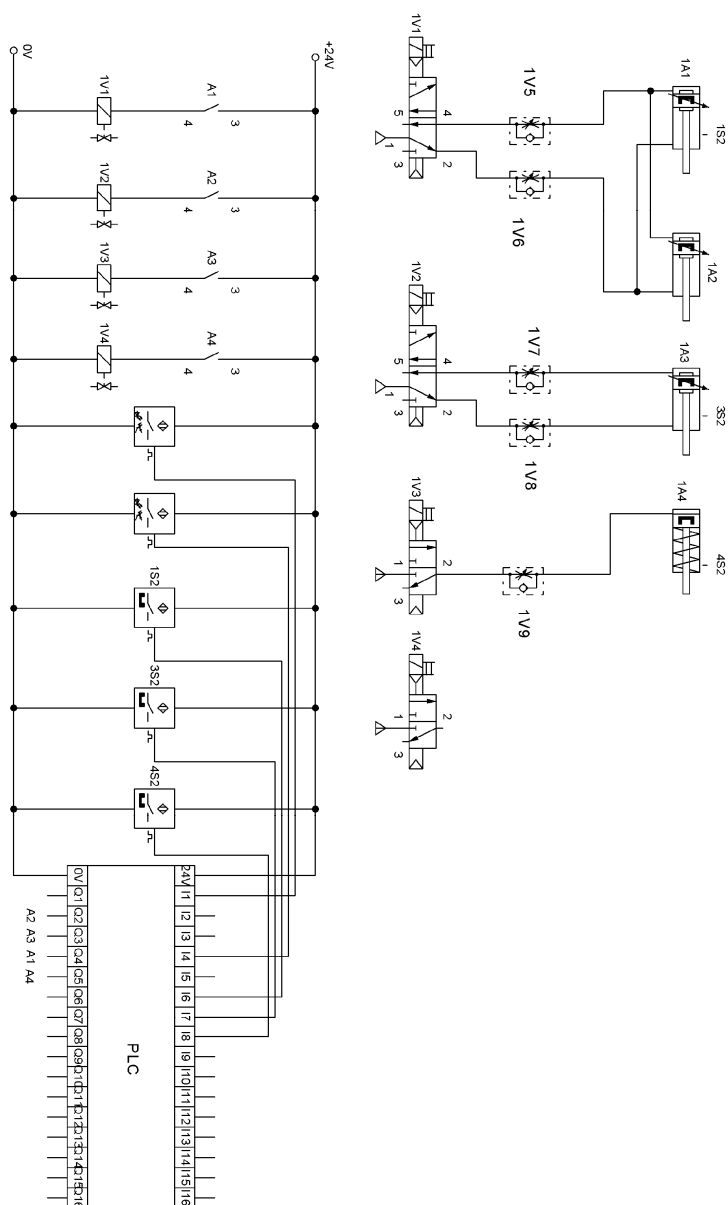
Příloha 4 Uchycení kamery InSight1403



Příloha 5 Simulace pneumatických pohonů manipulátoru



Příloha 6 Schéma elektropneumatické části manipulátoru



Příloha 7: Program detekce vad součástek

Image								Nalezen znaku soucastky:			
Index	Row	Col	Angle	Scale	Score						
Patterns	0.000	391.238	344.039	-1.469	100.000	85.977	1.000	1.000			
Uprava obrazu:								0.000			
Rozdil:											
Image											
Diskretizace:		Filtre:		Uprava pro nalezeni ryhy:							
Image		Image		Image							
Hledani ryhy											
	Row0	Col0	Row1	Col1	Score		Row0	Col0	Row1	Col1	Score
Edges	#ERR	#ERR	#ERR	#ERR	0.000		Edges	#ERR	#ERR	#ERR	0.000
	Row0	Col0	Row1	Col1	Score	Vyhodnoceni:					
Edges	#ERR	#ERR	#ERR	#ERR	0.000	0.000	0.000	0.000	0.000		
Osetreni pritomnosti plochy											
	Row0	Col0	Row1	Col1	Score						
Edges	#ERR	#ERR	#ERR	#ERR	0.000						
harna pritomna		0.000									
pocet otoceni		120.000		1.000	1.000						
Hledani otlaku typu 1:											
	Index	Row	Col	Angle	Color	Score	Area	Elongation	Holes	Perimeter	Spread
Blobs	0.000	#ERR	#ERR	#ERR	#ERR	0.000	#ERR	#ERR	#ERR	#ERR	#ERR
	Index	Row	Col	Angle	Color	Score	Area	Elongation	Holes	Perimeter	Spread
Blobs	0.000	#ERR	#ERR	#ERR	#ERR	0.000	#ERR	#ERR	#ERR	#ERR	#ERR
vyhodnoceni zda je pritomen otlak:				0.000	0.000						
Otlak pritomen		0.000									
Hledani otlaku typu 2:											
	Row0	Col0	Row1	Col1	Score						
Edges	#ERR	#ERR	#ERR	#ERR	0.000	0.000					
	Row0	Col0	Row1	Col1	Score						
Edges	#ERR	#ERR	#ERR	#ERR	0.000	0.000					
Vysledek analyzy											
0.000		0.000									

Příloha 8: Program kontroly rozměrů

Image									
Kalibrační data				Odchylka					
	Distance	Score			Distance	Score			
Edges	0.000	0.000	0.000 mm	Edges	0.000	0.000	0.000 mm	0.000 mm	0.000 ●
	Distance	Score			Distance	Score			
Edges	0.000	0.000	0.000 mm	Edges	0.000	0.000	0.000 mm	0.000 mm	0.000 ●
	Distance	Score			Distance	Score			
Edges	0.000	0.000	0.000 mm	Edges	0.000	0.000	0.000 mm	0.000 mm	0.000 ●
	Distance	Score			Distance	Score			
Edges	0.000	0.000	0.000 mm	Edges	0.000	0.000	0.000 mm	0.000 mm	0.000 ●
	Distance	Score			Distance	Score			
Edges	0.000	0.000	0.000 mm	Edges	0.000	0.000	0.000 mm	0.000 mm	0.000 ●
	Distance	Score			Distance	Score			
Edges	0.000	0.000	0.000 mm	Edges	0.000	0.000	0.000 mm	0.000 mm	0.000 ●
	Distance	Score			Distance	Score			
Edges	0.000	0.000	0.000 mm	Edges	0.000	0.000	0.000 mm	0.000 mm	0.000 ●
	Distance	Score			Distance	Score			
Edges	0.000	0.000	0.000 mm	Edges	0.000	0.000	0.000 mm	0.000 mm	0.000 ●
Kalibruj									